Personal Computer

MZ-2000

BASIC／MONITOR MANUAL

SHARP

Personal Computer

MZ-2000

BASIC／Monitor Manual

BASIC編 ········P.3～P.102

MONITOR編 ·····P.103～P.206

ご　注　意

このマニュアルのBASIC編は、パーソナルコンピュータMZ-2000のシステムソフトウェアBASICインタープリタMZ-1Z001、およびMONITOR編は同じくMZ-2000のシステムソフトウェアMONITOR MZ-1Z001Mにそれぞれ基づいて作成されています。

(1) 多目的パーソナルコンピュータMZ-2000では、システムソフトウェアをすべてファイル形態のソフトウェアパック（カセットテープ、ディスケットなど）によってサポートされます。各システムソフトウェアおよび本書の内容は、改良のため変更することがありますので、ファイルバージョンナンバーには特にご注意されるよう、お願い致します。

(2) 本書は、誤り、記載もれ等のないよう確認、チェックを行った上で作成しましたが、もしお気付きの点がありましたら、最寄りのシャープのサービス窓口（技術サービス部・サービスステーション・サービスブランチ）までご連絡願います。

(3) パーソナルコンピュータMZ-2000のシステムソフトウェアは、すべてシャープ株式会社のオリジナルソフトウェアであり、著作権は当社が保有しております。システムソフトウェアならびに本書の内容を無断で複製することは禁じられています。

(4) 本書のプログラムを使用したことによる金銭上の損害および逸失利益または第三者からのいかなる請求についても当社はその責任を負いませんのであらかじめご了承ください。

BASIC編

BASIC編目次

はじめに…… 8

第1章　BASIC MZ-1Z001の概要 …… 9

1.1　BASIC インタープリンタMZ-1Z001の起動 …… 10
1.2　動作モード…… 11
　1.2.1　コマンドレベルでの動作…… 11
　1.2.2　ステートメントレベルでの動作…… 12
1.3　BASIC プログラムの構造 …… 13
　1.3.1.　行…… 13
　1.3.2　文…… 13
1.4　BASIC文の構成要素…… 14
　1.4.1　予約語…… 14
　1.4.2　データ要素…… 15
　1.4.3　定　数…… 15
　1.4.4　変　数…… 16
　1.4.5　配　列…… 17
　1.4.6　式…… 18
　1.4.7　セパレータ…… 19
1.5　スクリーンエディタ…… 20
1.6　初期設定値について…… 22

第2章　BASIC MZ-1Z001のコマンド …… 23

2.1　プログラムファイル入出力コマンド…… 24
　2.1.1　LOAD …… 24
　2.1.2　SAVE …… 25
　2.1.3　VERIFY …… 26
2.2　テキスト編集コマンド…… 27
　2.2.1　AUTO …… 27
　2.2.2　LIST …… 27
　2.2.3　LIST/P …… 28
　2.2.4　NEW…… 28
2.3　コントロールコマンド…… 29
　2.3.1　RUN …… 29
　2.3.2　CONT …… 29
　2.3.3　MON …… 30
　2.3.4　BOOT …… 30

2.4　デファイナブルファンクションキー・リスト・コマンド……31
2.4.1　KLIST……31

第3章　BASIC MZ-1Z001のステートメント……33

3.1　代入文……34
3.1.1　LET……34
3.2　入出力文……35
3.2.1　PRINT……35
3.2.2　INPUT……38
3.2.3　GET……39
3.2.4　READ～DATA……39
3.2.5　RESTORE……40
3.3　ループ文……41
3.3.1　FOR～NEXT……41
3.4　分岐文……43
3.4.1　GOTO……43
3.4.2　GOSUB～RETURN……43
3.4.3　IF～THEN……44
3.4.4　IF～GOTO……46
3.4.5　IF～GOSUB……46
3.4.6　ON～GOTO……47
3.4.7　ON～GOSUB……47
3.5　定義文……48
3.5.1　DIM……48
3.5.2　DEF FN……49
3.5.3　DEF KEY……50
3.6　注釈文とコントロール文……51
3.6.1　REM……51
3.6.2　STOP……52
3.6.3　END……52
3.6 4　CLR……52
3.6.5　CURSOR……53
3.6.6　CSRH……53
3.6.7　CSRV……53
3.6.8　CONSOLE……54
3.6.9　CHANGE……57
3.6.10　REW……57
3.6.11　FAST……58
3.6.12　SIZE……58
3.6.13　TI$……58
3.7　音楽コントロール文……59
3.7.1　MUSIC……59

3.7.2 TEMPO……61
3.8 グラフィックコントロール文……62
3.8.1 GRAPH……62
3.8.2 SET……63
3.8.3 RESET……64
3.8.4 LINE……65
3.8.5 BLINE……66
3.8.6 POSITION……67
3.8.7 PATTERN……68
3.8.8 POINT……70
3.8.9 POSH……71
3.8.10 POSV……71
3.9 データファイル入出力文……72
3.9.1 WOPEN/T……72
3.9.2 PRINT/T……72
3.9.3 CLOSE/T……73
3.9.4 ROPEN/T……74
3.9.5 INPUT/T……74
3.10 機械語プログラムコントロール文……75
3.10.1 LIMIT……75
3.10.2 POKE……76
3.10.3 PEEK……76
3.10.4 USR……77
3.11 プリンタ・コントロール文……78
3.11.1 PRINT/P……78
3.11.2 IMAGE/P……78
3.11.3 COPY/P……79
3.11.4 PAGE/P……79
3.12 I/Oポートアクセス文……80
3.12.1 INP……80
3.12.2 OUT……80

第4章 BASIC MZ-1Z001の関数……81

4.1 組み込み数値関数……82
4.1.1 ABS……82
4.1.2 INT……83
4.1.3 SGN……83
4.1.4 SQR……84
4.1.5 SIN……84
4.1.6 COS……85
4.1.7 TAN……86
4.1.8 ATN……86

4.1.9 EXP……87
4.1.10 LOG……88
4.1.11 LN……88
4.1.12 RND……88
4.2 ストリング処理関数……89
4.2.1 LEFT＄……89
4.2.2 MID＄……89
4.2.3 RIGHT＄……90
4.2.4 SPACE＄……90
4.2.5 STRING＄……91
4.2.6 CHR＄……91
4.2.7 ASC……92
4.2.8 STR＄……92
4.2.9 VAL……93
4.2.10 LEN……93
4.2.11 CHARACTER＄……94
4.3 タビュレーションコントロール関数……95
4.3.1 TAB……95

付 録……97

A.1 ASCIIコード表……98
A.2 エラーメッセージ表……100
A.3 メモリマップ……102
A.4 三角関数・双曲線関数……102

MONITOR編目次……104

はじめに

このマニュアルは、パーソナルコンピュータMZ-2000の標準システムソフトウェアBASICインタープリタMZ-1Z001の言語仕様と文法を解説しています。

プログラミング言語BASICは、Beginner's All-purpose Symbolic Instruction Codeの頭文字であるといわれているように、各種のコンピュータ言語のうちで特に初心者向きの汎用プログラミング言語として開発されました。そして、多様な問題の解決にあたって、簡便に、しかもプログラマがコンピュータと対話する型式でプログラミングを進めることができるという特徴を持っており、パーソナルプログラミングにうってつけの、効率のよいプログラミング言語だといえます。

BASIC MZ-1Z001は、パーソナルコンピュータMZ-2000の特徴を充分に発揮させるBASICインタープリタであり、各種のアルゴリズム設計、データ処理、出力表示プロセス等にそれぞれ高機能を実現しています。同時に、高速処理を指向した設計がなされており、初心者だけでなく、高度な各種のプロフェッショナル・プログラミングにも縦横に駆使できるインタープリタです。

本書は、BASIC MZ-1Z001の言語仕様、コマンドグループ、ステートメントグループ、ファンクショングループの順に章を追って解説しています。

BASICインタープリタMZ-1Z001は、グラフィック3画面のコントロールを含んでいますが、カラーディスプレイコントロールは含んでいません。カラーディスプレイを接続して使用される場合は、カラーコントロール用BASICインタープリタ（別売：詳しくは販売店にお問い合わせください。）をご使用ください。

BASIC MZ-1Z001の概要

Chapter 1

本章は、BASICインタープリタMZ-1Z001の使い方、プログラミングの概要を解説しています。
はじめに、BASICインタープリタMZ-1Z001の起動方法が説明され、つづいて、動作モード、プログラムの構造、プログラムの構成要素、スクリーンエディション、初期設定値の解説が行なわれます。
BASICインタープリタMZ-1Z001に用意されている全てのコマンド、ステートメント、関数は、つづく第2章、第3章、第4章で個々に詳細な文法解説がなされています。

1.1　BASIC インタープリタMZ-1Z001の起動

BASIC MZ-1Z001は、MONITOR MZ-1Z001Mとともにカセットテープファイル中にあり、その使用に際してイニシャルローディングを行う必要があります。これは難しいことではなく、MZ-2000の電源をONにした時、イニシャルローディングを行うIPL(initial program loader)が自動的にスタートするので、あらかじめ、BASICカセットファイルをカセットデッキにセットしておきます。テープをセットしていない場合は、テープをセットするようメッセージが表示されるのでその指示に従います(テープは巻き戻していること)。イニシャルローディングによって、MONITOR MZ-1Z001MおよびBASIC MZ-1Z001がシステムにローディングされ（図1－1の左の写真はローディング中であることを示しています)、終了後図1－1の右の写真に示すメッセージを表示し、BASICが起動します。

"Ready"は、システムコントロールがBASICのコマンドレベルにあり、コマンドを受け付けられる状態にあることを示しています。

（ソフトウェアの起動方法の詳細については、Owner's Manualの「システムプログラムのイニシャルローディングの方法」を参照のこと。)

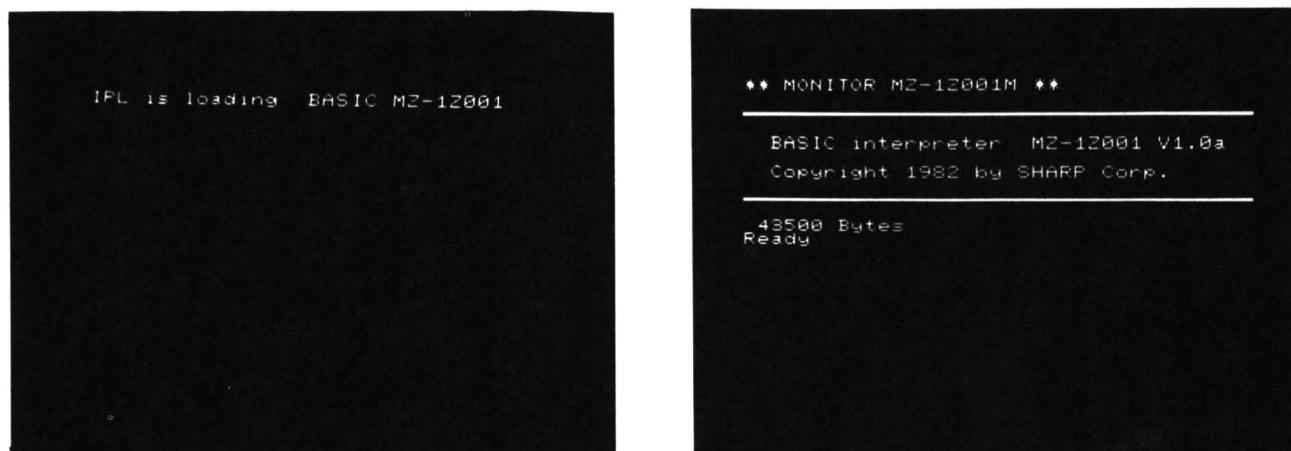



図 1—1

1.2 動作モード

システムソフトウェアBASICインタープリタの動作には、コマンドレベルの動作モードとステートメントレベルのプログラム実行モードの2通りの動作モードがあります。コマンドレベルの動作モードでは、BASIC コマンド を直接に実行したり BASIC プログラムテキストの作成・編集を行います。ステートメントレベルのプログラム実行モードでは、BASIC コマンド RUN によって BASIC プログラムテキストを解釈し、その指示に従って各種の自動処理を行います。

1.2.1 コマンドレベルでの動作

BASICのコマンドレベルの動作モードには　直接モード（*direct mode*）と間接モード（*indirect mode*）の2通りがあります。

直接モードでは、BASICの直接実行コマンド、たとえば、RUN、LIST、SAVE といったコマンドの実行、あるいは、直接実行命令としても使用できるステートメントを用いて、ちょうど電卓を使用するような使い方が可能です。

電卓のような使い方としては、たとえば図1－2に示すように、PRINT 文と、算術式によって、演算・表示コマンドを与えて、結果を直ちに得ることができます。

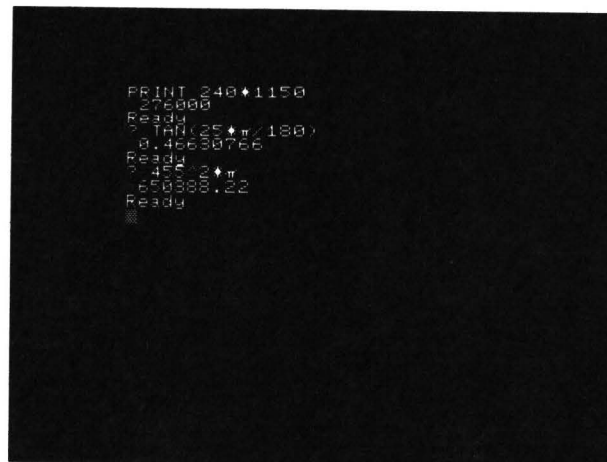



図 1—2

また、このような使い方では、プログラムの実行中、STOP文やBREAK 操作によって中断させ、その時の各変数の値を調べたり、値を代入したりするのが可能で、デバッグ操作として使うことができます。

間接モードとは、BASICプログラムの作成、編集（エディション）を行うモードです。BASICプログラムは、次項から詳しく解説されるように、行番号をもつBASIC文の集合であり、BASICに用意されているステートメントや関数によって目的とする処理のアルゴリズムを記述するものです。間接モードではカーソルエディションによってCRTディスプレイを見ながら編集作業を行うことができます。（1・5節を参照）

1.2.2　ステートメントレベルでの動作

プログラムのRUNコマンドによって実行モードへ移るとBASICインタープリタは、プログラムテキストに記述された各ステートメントを解釈しながら各々の機能にしたがったデータ処理を実行して行きます。実行モードにおけるプログラマとのコミュニケーションは、各種入出力文によって行われます。

たとえば、INPUT文の実行によって、必要なデータをキーボードから入力したり、PRINT 文の実行によって内部データをCRTディスプレイ上に出力したりのコミュニケーションがステートメントレベルによって行われます。

プログラムの実行を終了すると、動作モードは、コマンドレベルに戻ります。実行モードにある場合強制的にコマンドレベルに戻すには BREAK + SHIFT キーを押し、プログラム実行を停止させます。なお、BASICには、USR関数による機械語ルーチンの使用が可能ですが、この場合の動作モードは、CPUレベルの実行モードとなり、BASICの管理を離れることになります。もし、CPUレベルの実行モードにある場合、それを停止させるには、リセットスイッチ（本体背部にあります）を押し、MONITORコマンドレベルへ戻すしかありません。

1.3 BASICプログラムの構造

BASICプログラムは、何行かの順番に並べられた行（*line*）からなっており、それぞれの行には、プログラム実行、手続きなどの動作（*operation*）を指示する文（*statement*）が記述されます。

BASICプログラムの実行は、分岐命令、参照命令等を除いて、原則的に文の並びの順に行なわれます。またBASICでは、特別なフォーマット（*format*）指定文や宣言文などの初期設定項目からプログラムを始める必要はなく、プログラム中の任意の行から実行を開始することも可能です。

1.3.1 行 (line)

行は、行番号と本文とからなり、それぞれの行は、キャリッジリターンコードをセパレータとします。間接モードにおいて1行を入力するには CR キー（または ENT キー）を与えることによって実行されます。

行番号

行番号は、行の冒頭に置き、行を示す役割を果します。行を定義するという意味で、定義行番号（*definition line number*)ともいい、他の文から参照するために記述する参照行番号(*reference line number*)と区別して言うことがあります。

定義行番号は、1～65535 の範囲の整数が許されます。各行の定義行番号は、連続した数である必要はないので、行の追加挿入を考えて、10 ぐらいずつあけて設定するのが得策です。またAUTOコマンド（P. 27 参照）は一定の増分で定義行番号を自動的に発生してくれます。

同一の定義行番号の行が入力された場合は、以前のものは無効となります。

本　文

本文は、1つの行のなかで、定義行番号に続いて1つの文または、複数の文を記述します。

本文は、定義行番号と、キャリッジリターンコードを含めて1行、即ち160 桁以内となるような文字列で記述します。

本文を、複数文（*multi statement*）で記述する場合、文と文のセパレータとしてコロン" : "を使用します。

1.3.2 文 (statement)

BASIC文は、大きく実行文(*executable statement*)と非実行文(*non-executable statement*)とに分類できます。

実行文はプログラムの動作を記述するもので各種の演算や代入、比較、分岐を処理する文があります。非実行文はプログラムに必要な情報の設定やポインタのコントロールを行うもので、配列宣言文やデータ文、定義文、注釈文などがあります。

直接モードで使用する場合記述する文は、直接実行文（*direct mode executable statement*)と呼ぶこともあります。

1.4 BASIC文の構成要素

BASIC 文は、予約語であるコマンド、ステートメント、組み込み関数、システム変数だけでなく、各種の演算子、定数、変数名、配列名、式といった各要素とともに構成され ます。以下、各構成要素について解説を行います。

1.4.1 予約語(reserved words)

すべての予約語 (キーワードとも呼ばれます)は、BASIC インタープリタ自身が管理している語であり、プログラマが任意に定義することができない語の集まりです。表1－1は、BASICインタープリタMZ-1Z001の全ての予約語をアルファベット順に並べたものです。これらは、コマンド、ステートメント、組み込み関数、システム変数に対応した独自の働きがあり、プログラマが変数名などに使うことはできません。(表1－1の予約語の右の数字は、参照ページを示しています。)

	予約語	ページ		予約語	ページ		予約語	ページ
A	ABS	82		INP	80		PRINT/T	72
	ASC	92		INPUT	38	R	READ	39
	ATN	86		INPUT/T	74		REM	51
	AUTO	27		INT	83		RESET	64
B	BLINE	66	K	KLIST	31		RESTORE	40
	BOOT	30	L	LEFT $	89		RETURN	43
C	CHANGE	57		LEN	93		REW	57
	CHARACTER $	94		LET	34		RIGHT $	90
	CHR $	91		LIMIT	75		RND	88
	CLOSE/T	73		LINE	65		ROPEN/T	74
	CLR	52		LIST	27		RUN	29
	CONSOLE	54		LIST/P	28	S	SAVE	25
	CONT	29		LN	88		SET	63
	COPY/P	79		LOAD	24		SGN	83
	COS	85		LOG	88		SIN	84
	CSRH	53	M	MID $	89		SIZE	58
	CSRV	53		MON	30		SPACE $	90
	CURSOR	53		MUSIC	59		SQR	84
D	DATA	39	N	NEW	28		STEP	41
	DEF FN	49		NEXT	41		STOP	52
	DEF KEY	50	O	ON	47		STR $	92
	DIM	48		OUT	80		STRING $	91
E	END	52	P	PAGE/P	79	T	TAB	95
	EXP	87		PATTERN	68		TAN	86
F	FAST	58		PEEK	76		TEMPO	61
	FOR	41		POINT	70		THEN	44
G	GET	39		POKE	76		TI $	58
	GOSUB	43		POSH	71		TO	41
	GOTO	43		POSITION	67	U	USR	77
	GRAPH	62		POSV	71	V	VAL	93
I	IF	44		PRINT	35		VERIFY	26
	IMAGE/P	78		PRINT/P	78	W	WOPEN/T	72

表 1−1 BASICインタープリタMZ－1Z001の全ての予約語

1.4.2 データ要素

BASICプログラムで予約語以外の構成要素は一般にデータ要素と呼ぶことができます。それらを分類すると更に次のような各要素に分けられます。

定　　数	：	数値定数、ストリング定数、システム定数
変　　数	：	数値変数、ストリング変数、システム変数
配　　列	：	1次元数値配列、2次元数値配列、1次元ストリング配列、2次元ストリング配列
式	：	算術式、ストリング結合式、関係式、論理式
セパレータ	：	コンマ(，)、セミコロン(；)、コロン(：)

定数、変数、配列の各データ要素には、上記のようにデータの型式（*type*）によって、数値データとストリングデータの2通りのものがあります。

1.4.3 定　　数

定数とは、その表現自体がデータの値を具体的に示すものであり、プログラムの実行中にその値が変えられることのないデータ要素をいいます。

数値定数（numeric constant）

数値定数は符号（＋、－）、数字（0～9）、小数点（ ． ）で記述される10進数、あるいは符号、仮数部、指数部（Eで示す）で記述される指数表現の 10 進数をいいます。いずれも内部表現型式は浮動小数点型式をとります。

BASICインタープリタMZ-1Z001で表現できる数値データは、有効桁8桁以内（精度は最小桁±1以内）、指数範囲、10^{-19}～10^{19}です。

数値が正の場合は、＋符号は省略できます。

LIMIT、POKE、PEEK、USRの各文においてアドレスを直接指定する場合あるいは CHR＄ によってアスキーコードを指定する場合は、16進数を用いることも可能です。16進4桁（CHR＄の場合は2桁）の前にドルマーク"＄"をつけて示します。　例：LIMIT　＄B000

（正　例）	5215E－8 ＝0.00005215	
（誤　例）	15,300	コンマを使っている
	123456789	8桁を越えている
	300E＋91	指数範囲を越えている

ストリング定数（string constant）

ストリング定数は、キーボードから入力可能な文字セットの並びによって表現する文字定数であり、一対のクォーテーションマーク（"）で囲んで示します。DATA文中では、クォーテーションマークは省略できます。

ストリング定数の最大文字数は、行の有効長によって決まりますが、BASIC内部でのストリングデータの最大文字数は255文字です。

ストリング定数は、PRINT文ではメッセージや各種キャラクタを表現し、MUSIC文では音符データを表現し、PATTERN文では、グラフィックパターンのビットデータを表現するなど、ステートメントによってそれぞれ異った各種のデータを表わします。

システム定数(system constant)

システム定数は、BASICに用意されている定数であり、円周率の値が、キャラクタ"π"に定められています。π＝3.1415927

(例)　半径が10の円の円周は、次の式によって計算できます。

2＊π＊10

1.4.4 変　数(variable)

変数とは、データの型は変わらないが、その値はプログラム実行中に任意に変えることのできるものをいいます。変数は、変数名によって引用されます。変数の初期値は、0または空(ブランク)です。

数値変数(numeric variable)

数値変数には数値データだけを与えることができます。

変数名は何文字でもかまいませんが、最初の2文字だけが変数の区別に使われます。また最初の文字は英字の大文字(A～Z)でなければなりませんが残りの文字は、英字と数字が使えます。ただし、BASICで使用している予約語を使用することはできません。予約語(*reserved word*)とは、BASICの全てのコマンド、ステートメント、関数名、および演算子、その他の特殊文字(INP、OUT文で使う@マークやMUSIC文で使う#マークなど)です。

数値変数は数値データが代入されるまでは0となっています。

(正　例)	ABCとAB	は同じ扱い
(誤　例)	DATA3	予約語DATAを含む
	C@	特殊文字を含む

ストリング変数(string variable)

ストリング変数にはストリングデータだけを与えることができます。

変数名は、前記、数値変数名の場合の条件に従い、名前の最後にドルマーク"$"を付加してストリング変数名とします。ストリング変数が扱いうるストリングデータの最大長は255文字です。ストリング変数は、ストリングデータが代入されるまではブランクとなっています。

(正　例)　NAME1$とNAME2$は同じストリング変数とみなす

(誤　例)　music$　………小文字を使っている。

システム変数（system variable）

システム変数は、BASIC内で変化する値を示す変数であり、BASICの空きエリアを示すSIZE、カーソル位置を示すCSRH，CSRV，グラフィックエリア上のポジションポインタの位置を示すPOSH，POSVの各数値変数、および、内蔵の24時間時計の値を示す6桁のストリング変数TI$があります。

1.4.5 配　列（array）

配列は、順序づけられた同じ型のデータの集合であり、添字を1個もつ1次元配列(リストともいう)と、添字を2個もつ2次元配列(テーブルともいう）があります。

数値配列とストリング配列のディメンジョン定義等の詳細は、文法編3・5・1節（P. 48　）に解説されています。

1.4.6 式（expression）

式には前述のように算術式、ストリング結合式と関係式、論理式とがあります。算術式とストリング結合式は、ステートメントのオペランドの中で、また関数の引数として使い、式に従った演算の結果である１つの数値データまたはストリングデータを表わします。関係式と論理式は、ＩＦ文のオペランドで分岐条件を記述する際に用いられます。

算術式（arithmetic expression）

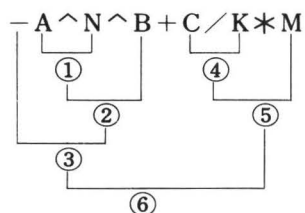
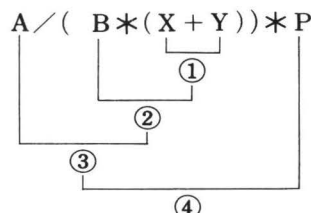
算術式は、数値データ（数値定数、数値変数、数値配列、数値関数）の算術演算を表現するものであり、数値データである算術要素と、算術演算子とから構成されます。

算術演算子は、四則演算子と、べき算演算子とからなり、それらを演算の優先順に並べると次のようになります。

算術演算子	演　算	形　　式
^	べき算	X^Y（X^Y：XをY乗する）
－	負符号	－X　（Xに－１を掛ける）
＊，／	乗算、除算	X＊Y（XにYを掛ける）、X／Y（$\frac{X}{Y}$：XをYで割る）
＋，－	加算、減算	X＋Y（XにYを足す）、X－Y（XからYを引く）

べき算が優先されますが、括弧"（　）"を使用している場合は、括弧内の演算がさらに優先されます。括弧内の演算の優先順も上記と同様です。

＊と／や、＋と－のように、同順位の演算が続く場合は、式の左から右へ順に計算されます。また、べき乗が連続する場合も同様です。

たとえば、次に示すように演算順が決まります。

即ち、　$-(A^N)^B+\frac{C}{K}M$

即ち、　$\frac{A}{B(X+Y)}P$

(例)　いくつかの一般の数式をBASICの算術式で表現してみます。

一般の数式	BASICの算術式
$\frac{c}{a+b}-\frac{e}{d}$	C／（A＋B）－E／D
$\frac{c}{a\,b}$	C／（A＊B）またはC／A／B
$\sin^2 x+1$	SIN（X）^2＋1
x^{y^z}	X^（Y^Z）
$(x^y)^z$	X^Y^Z
$\tan^{-1} x$	ATN（X）
$2\pi r$	2＊π＊R
$\sqrt{a^2+b^2}$	SQR（A^2＋B^2）または（A＊A＋B＊B）^0.5

ストリングの結合式（string connective expression）

ストリングの結合式とは、文字通り幾つかのストリングデータを連結して新たな１つのストリングデータを作る式であり、演算子"＋"を用いて表記します。

(例)　"ＡＢＣ"＋"ＤＥＦ"　は　"ＡＢＣＤＥＦ"と同じストリングデータを表わします。

次のプログラムを実行すると、ＡＬ＄にはアルファベット26文字が代入されます。

プログラム

```
10 FOR N=1 TO 26
20 AL$=AL$+CHR$ (64+N)   ……………ストリングの結合式が使われています。
30 NEXT N
40 PRINT AL$
50 END
```

プログラムの実行

```
RUN
ABCDEFGHIJKLMNOPQRSTUVWXYZ
Ready
```

関係式（relational expression）

関係式は、２つの数値データを関係演算子によってつないだもの、または、２つのストリングデータを関係演算子によってつないだものです。

関係式はＩＦ文の中で使用されるので、詳細は文法編３．４．３（P.44）に解説されています。

論理式（logical expression）

論理式は、関係式によって与えられる真または偽の値の論理和または論理積を表現するものであり、ＩＦ文の中で使用されます。

1.4.7 セパレータ（separator）

１つの文の中で幾つかの要素を区切る場合は、セパレータとしてコンマ"，"を使用します。

(例)　ＤＡＴＡ　３.５，４.６６，ＤＡＴＡ１

　　　ＯＮ　Ａ　ＧＯＴＯ　１００，２００，３００

1.5　スクリーンエディタ

間接モードでBASICプログラムを作成、編集する時はBASICのスクリーンエディタが動作しており、画面を見ながらカーソル（*cursor*）を自由に動かして、大変操作性のよいエディションが可能となっています。

BASICプログラムは、行番号をもつ行の単位でBASICテキストエリアにストアされますが、すでに述べたようにキャリッジリターンキー CR を入力するか、あるいは、エンターキー ENT を入力することによってはじめてストアされます。従ってプログラムのリスト表示をカーソル操作によって変更した場合も、必ず CR キーか ENT キーを入力する必要があります。

カーソル操作（cursor control）

プログラムを新しく入力する場合は、行番号、本文の順に行をキー入力して行き、その場合、カーソルは、1文字分ずつ移動して行きます。

一方、既にストアされているプログラムのある行を変更する場合は、その行のリスト表示上にカーソルを持って行き変更することが可能であり、行全体を新しく入力し直す必要はありません。カーソルコントロールは、左右上下に動かすための4個の黄色のキーと、カーソルを画面の左上隅に移す CLR HOME キーによって行います。

左右上下移動のキーは、SHIFT キーと共に押すと、オートリピートが可能です。

行の変更の場合、単にある文字を他の文字に変えるだけでなく、挿入や削除が必要な場合があります。挿入を行う場合は SHIFT キーを押したまま必要回 INST DEL キーを入力すると、カーソル位置にスペースができます。削除を行う場合は、必要回 INST DEL キーを入力すると、カーソル位置の手前を削除して行きます。

図　1—3

1つの行全体を削除する場合は、その行の行番号のみ入力し CR または ENT キーを入力することにより行なわれます。

BASIC テキストは、通常プログラムの実行、あるいは実行中のエラー検出、ブレーク等によって壊されることはありません。プログラム上の間違いは、実行によって見つけ出し、スクリーンエディションによって必要なデバッグ操作を繰り返すことによって排除して行くことができます。このような対話型のデバッグ・エディション操作はBASICの大きな特徴といえます。

タブキー TAB のセッティング

メインキーボード上で、スペースバーのすぐ左側にある TAB キーは、タビュレーションキー（*tabulation key*）で、キー入力の際に指定欄に一定の書式でデータを入力する場合等に使われます。

タブセットは、モニタワークエリアの＄11Ｃ1から＄11ＣＦの間のデータを変えることによりプログラマが必要とするものに設定し直すことができます。

この操作は、MONITOR コマンドによって行うこともできますが、ここでは、BASICのＰＯＫＥ文を使ったタブセットの方法を示します。

タブセットのデータエリアは、＄11Ｃ1番地～＄11CF番地の間の15バイトであり、＄11C1 番地に最初のタブセット位置１を(キャラクタ表示のＸ－座標)、つづけて第２、第３のタブセット位置を入れてゆきます。$11C0番地には１をストアしておきます。

たとえば、最初のタブセット位置をＸ－座標が15の位置（即ち16キャラクタ目）とするには、

```
POKE  $11C0，1
```

の実行後に、

```
POKE  $11C1，15
```

を実行することによってセットできます。続いて、次のタブセットを、Ｘ－座標が２3の位置とするには、

```
POKE  $11C2，23
```

を実行します。

ここで次は改行して行の先頭へ戻すには、＄11Ｃ3番地にタブセットのエンドマーク２５５（＄ＦＦ）をセットしておきます。

```
POKE  $11C3，255
```

を実行します。

1.6　初期設定値について

MZ-2000システムIPLによって、BASICインタープリタMZ-1Z001を起動した時の各種初期設定値をまとめて示します。これらはシステムの初期デフォルト値であり、いずれの値もプログラマが任意に変えることができます。

■キーボード関係

1)　動作モード：ノーマルモード
2)　小文字の入力はノーマルモードのシフトポジション
3)　デファイナブルファンクションキーは最初、次のように設定されます。

F1 ：RUN ↵
F2 ：LIST↵
F3 ：CONSOLE
F4 ：CONT↵
F5 ：AUTO↵
F6 ：CHR$(
F7 ：DEF KEY (
F8 ：GRAPH
F9 ：SAVE"
F10 ：LOAD ↵

■CRTディスプレイ関係

1)　キャラクタディスプレイモード：　ノーマル（バックグラウンド：黒）
2)　キャラクタ表示桁数：　40キャラクタ／行
3)　キャラクタ表示スクローリングエリア：　最大（第0行から第24行）
4)　グラフィックディスプレイ入力モードページ：　ページ1
　　グラフィックディスプレイ出力モードページ：　全ページともOFF
　　ポジションポインタ：　POSH＝0、POSV＝0
5)　グラフィックディスプレイ、リゾリューションモード：　320×200ドット／画面

■内蔵時計

TI$＝"000000"で初期化してスタートする。

■音楽機能

1)　テンポのデフォルト値：　4（中庸のテンポ、Moderato）
2)　音長のデフォルト値：　5（4分音符、♪）

■その他

1)　配列変数は全て未定義
2)　BASICテキストエリアの下限：　$FFFF番地（即ちLIMIT MAX 状態）
3)　タブセット：　5キャラクタ毎

BASIC MZ-1Z001のコマンド

Chapter 2

この章は、BASICインタープリタMZ-1Z001の持つすべてのコマンドを説明します。コマンドは、特別のものを除いて、直接実行命令としてのみ使用できる命令であり、プログラムテキスト中で使用することはできません。

なお、MZ-80Bで作成されたBASICテキストプログラムは、MZ-2000のBASICインタプリタMZ-1Z001およびDISK BASICインタプリタMZ-2Z001のどちらでも使用可能です。ただしハードウェアに直接関係した命令（LIMIT、PEEK、POKE、USR、~~IN~~INPおよびOUT）をプログラム内で使用している場合には、修正を要します。(BASICテキストエリアおよび各ポートアドレスがMZ-80BとMZ-2000とでは異なる為）またPOINT命令については3.8.8 (P.70)を参照ください。

各コマンド解説の形式

書　式　コマンドの正しい一般形を示します。
ここで、プログラマが指定すべき項目は、イタリック体で示しています。
〈 〉内に示した項目は、省略可能であるか任意に繰り返し項目を連ねることができることを示しています。
項目のセパレータ（コンマ、セミコロンなど）は正しく決められた位置に置かなければなりません。

機　能　コマンドの機能を述べます。

解　説　コマンドの使い方を詳しく説明します。

例　コマンドの使い方の例を示します。

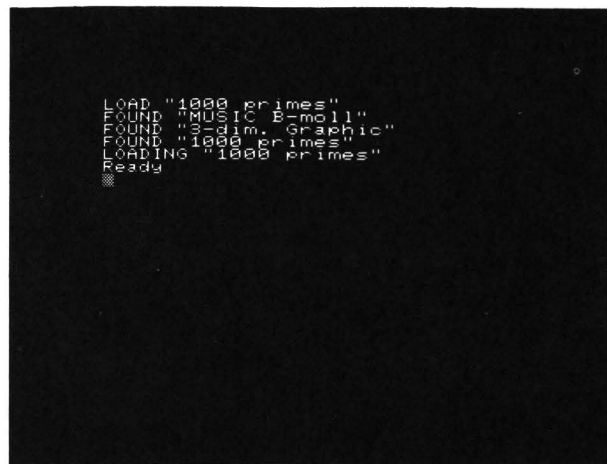
2.1　プログラムファイル入出力コマンド

2.1.1 LOAD

書　式　LOAD 〈*file name*〉

機　能　カセットテープファイルから、*file name* で指定するBASIC テキストファイル、またはBASIC プログラムにリンクされる機械語プログラムファイルをロードします。

解　説　ロードすべきファイル名はオペランドに記述します。ローディング動作の第一段階では、ファイルサーチが行われ、ファイルのヘッダ部に登録されているファイル名を読み、指定されたファイルであるかどうか調べます。 LOAD コマンドのオペランドに指定されたファイル名と異ったファイルは、

　　FOUND　"*file name*"

の表示を行った後、早送りで読みとばされます。指定されたファイルが見つかった時は、

　　FOUND　"*file name*"

に続けて、

　　LOADING　"*file name*"

が表示され、ローディングが行われます。図2－1は、2つのファイルを読みとばした後に指定ファイル"1000 primes" のローディングを行ったもようを示しています。

図　2―1

もしオペランドのファイル名指定を省略すると、最初に見つかったファイルのローディングを行います。

ファイルのローディングは、ファイルモードが BTX (BASICテキスト) またはOBJ (機械語プログラム) のものだけに実行されます。

ロードすべきファイルがBTX モードである場合、BASIC テキストエリアを NEW した後、新たにロードされることになり、前にあったテキストは消去されます。

OBJ モードのファイルのローディングは、LIMIT 文によって確保した機械語プログラムエリアが既にあり、しかも該当 OBJ モードファイルのローディングアドレスが、そのエリア内であればローディングが実行されます。この場合、BASIC テキストエリア内のテキストは消去されません。
OBJ モードファイルのローディングを行う場合、LOAD コマンドは、プログラム中のステートメントとして使用することができます。
OBJ ファイルのローディングアドレスは、MONITOR あるいは、LINKER 等のシステムプログラムのファイル出力時に指定し、ファイルのヘッダ部に登録されます。(BASICインタープリタには、OBJ ファイルを出力する機能はありません。)

2.1.2 SAVE

書　式　SAVE 〈*file name*〉

機　能　BASIC テキストエリアにストアされている BASICプログラムテキストをファイル名を与えてカセットテープ上にセーブします。

解　説　セーブされるテキストは、BASIC プログラムテキストの全部（即ち、LIST コマンドによって表示されるプログラムテキストの全体）であり、機械語プログラムエリアの内容はセーブの対象とはなりません。
file name は、作成される BASIC テキストファイル（BTX）の名前となるので、プログラムに合った適当な名前を16文字以内のストリングで与えます。*file name* を与えないと、エラーにはなりませんが、名なしのファイルが作成されることになり、そうしたファイルを多数作るとファイル管理が難しくなります。

例　現在 BASIC テキストエリアにストアされているプログラムテキストを、ファイル名を "Process 1－1" としてセーブするには、カセットテープをカセットデッキにセットし、

SAVE "Process 1－1" [CR]

を実行します。

2.1.3 VERIFY

書式　VERIFY 〈*file name*〉

機能　BASIC テキストエリア内のプログラムテキストと、*file name* で指定するBASIC テキストファイルの内容を比較します。

解説　両者の内容が同一であれば"ＯＫ"が表示され、異っている場合は"ERROR"表示がなされます。図２－２の左側は、BASIC テキストファイル"TEXT　cycloid"をテキストエリア内のテキストと比較して両者が一致した様子を（ここで、該当ファイルは、ファイル"MUSIC"の後に見つかったことが判ります）、右側の写真は、BASIC テキストファイル"MUSIC"の比較が成立しなかったことを示しています。

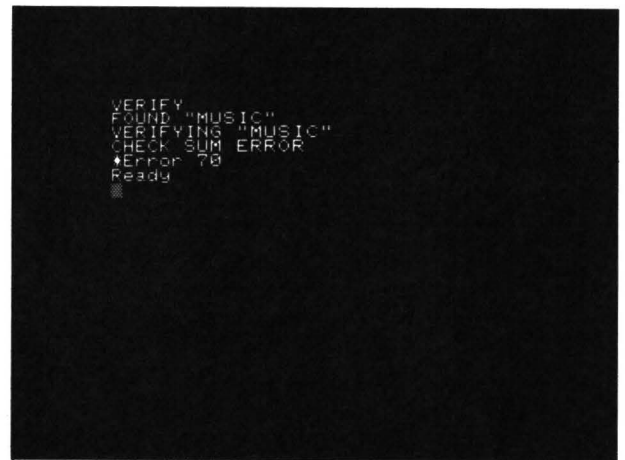
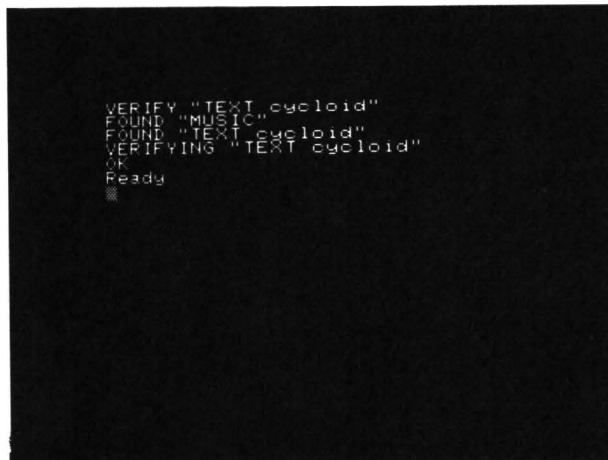



図　2−2

オペランドにファイル名を記述しないと、最初に見つかったBASIC テキストファイルが比較の対象となります。

2.2 テキスト編集コマンド

2.2.1 AUTO

書　式　AUTO　<*ls*, *n*>

ls………スタート行番号：デフォルト値=10

n………行番号の増分：デフォルト値=10

機　能　間接モードでのBASICテキスト作成時に、定義行番号を自動的に発生させます。

解　説　AUTOコマンドを実行すると、BASICは行番号*ls*を表示してステートメントの入力を待ちます。[CR]キー（または[ENT]キー）によってその行を入力すると、改行して次の行番号*ls*+*n*を表示してステートメントのキー入力を待ちます。

このように、AUTOコマンドは行番号*ls*からスタートして、一定の間隔*n*を置いた定義行番号を発生し、テキスト作成がスムーズに行えるようにしています。

オペランドのデフォルト値（指定しなかった場合に、BASICが決める値）は、*ls*、*n*ともに10です。

AUTOコマンドから通常の間接モードへ戻るには[BREAK]キーを押します。

例

AUTO 300,5………定義行番号を、300、305、310、315………の順に発生します。

AUTO 100………定義行番号を、100、110、120………の順に発生します。

AUTO ,20………定義行番号を、10、30、50、70………の順に発生します。

AUTO ………定義行番号を10、20、30、40………の順に発生します。

2.2.2 LIST

書　式　LIST　<*ls*-*le*>

ls………スタート行番号

le………エンド行番号

機　能　現在BASICテキストエリア内にあるプログラムテキストの全部、もしくは一部をCRTディスプレイ上にリストします。

解　説　プログラムテキストの全部をリストするにはオペランドは記述しません。

一部分をリストさせる場合は、リストスタート行番号*ls*とエンド行番号*le*をマイナス記号"－"で結ぶか、どちらか一方を記述します。

リスト表示の実行中にスペースバーを押さえると、その間リスト表示が一但停止します。また、また、[BREAK]キーを押すと、リスト表示が中断されてBASICコマンドレベルへ戻ります。

なお、このLIST命令を実行すると、グラフィックディスプレイ出力モードページは、全ページともOFFとなります。

例

LIST ………プログラムテキストの全体をリストします。

LIST 300-350………定義行番号300から350までのプログラムテキストをリストします。

LIST 2500-………定義行番号2500以降のプログラムテキストをリストします。

LIST 150………定義行番号150のテキストのみをリストします。

2.2.3 LIST/P

書　式　LIST/P　<*ls*－*le*>

ls………スタート行番号

le………エンド行番号

機　能　現在BASICテキストエリア内にあるプログラムテキストの全部、もしくは一部をオプションのラインプリンタ上にリストします。

解　説　プリンタ上にリストするプログラムテキストの範囲の指定方法は、前記LISTコマンドのオペランドの記述と同様です。

例　LIST/P………プログラムテキストの全体をプリンタ上へリストします。

PRINT/P　CHR＄（5）：LIST/P-2990

………プリンタのフォームフィード（form feed）を行った後、即ち、次ページの先頭に紙送りをした後に、プログラムテキストの行番2990までをプリンタ上へリストします。

プリンタの状態によって次のエラーが発生することがあります。

＊Error 65 ………プリンタがレディ状態（ready）でない

＊Error 66 ………プリンタのハードウェア上のエラー発生

＊Error 67 ………プリンタの用紙切れ

2.2.4 NEW

書　式　NEW

機　能　BASICテキストエリアをクリアすると共に、プログラムワークエリアの初期化を行います。

解　説　NEWコマンドを実行すると、BASICテキストエリア内にあるプログラムテキストを削除し、同時に変数エリア、配列エリア等のプログラムワークエリアを削除します。

LIMIT文によってBASICテキストエリアが制限されている場合は、、BASICテキストエリア内だけがクリアされ、機械語プログラムは削除されません。

LOADコマンドの実行時（BTXファイル）にもローディングの前に、NEWコマンドと同じ操作が行われます。

2.3　コントロールコマンド

2.3.1　RUN

書　式　RUN　<*ls*>

ls………スタート行番号

機　能　現在BASICテキストエリア内にあるプログラムを実行します。

解　説　間接モードのテキスト編集操作によって作成したプログラム、あるいはカセットテープファイルからロードしたプログラムを実行させるコマンドです。

プログラムを、その先頭行から実行するには単に、

RUN [CR]

とキー入力します。

行番号*ls*から実行する場合は、

RUN *ls* [CR]

とキー入力します。

オペランドを記述しないRUNコマンドでは、はじめにCLR文（3・6・4節参照）が実行されたのと同様に、変数エリアと配列エリアはクリアされ、その後、プログラムの先頭行へコントロールが移されます。一方、RUN *ls* コマンドでは、クリア操作は行われません。

システムコントロールは、プログラムの中のEND文、STOP文の実行あるいはプログラムの終端でコマンドレベルに戻ります。（途中でエラーが発生した場合や [BREAK] + [SHIFT] キーが押された場合もコマンドレベルに戻ります。）

また、プログラム中でUSR関数（3・10・4節参照）を使っている場合は、ユーザエリアの機械語プログラムやモニタサブルーチン等へ一旦コントロールが移されます。（この場合、コントロールはBASICの管理を離れることになり、[BREAK] キー操作等は受けつけなくなります。）

あるいは、MONコマンドをステートメントとして用いた場合は、それによってシステムコントロールはモニタコマンドレベルへ、BOOTコマンドをステートメントとして用いた場合はIPLスタート状態へコントロールが移されることになります。

2.3.2　CONT

書　式　CONT

機　能　プログラムの実行中の [BREAK] + [SHIFT] キーの入力、あるいはSTOP文、END文の実行によってプログラムが中断状態にある時、プログラムの実行を継続して再開します。

(CONT：continue)

解　説　CONTコマンドは、通常プログラムのデバッグのため用いられ、STOP文あるいは [BREAK] キーの入力によって任意の箇所で実行を停止させ、直接モードによって中間結果を調べたり、変数のデータを変えたりした後のプログラム実行の継続に使われます。

プログラムの再開を別の行から行う場合は、RUN *ls*コマンドまたは、GOTO *ls* コマンドを使います。

プログラム中断中に間接モードによるプログラムの編集作業（エディション）を行った場合はCONT不能(Error 17）となります。

2.3.3 MON

書　式　MON

機　能　BASICインタープリタMZ-1Z001によるシステムコントロールを離れて、MONITOR MZ-1Z001Mのシステムコントロールレベルへ移ります。（MON：MONITOR）

解　説　メモリマップ（付録A. 3）に示されるように、システムは、BASICインタープリタとMONITORの2層構造のシステムプログラムによって動作しています。IPLによってカセットテープからMONITOR MZ-1Z001MとBASIC MZ-1Z001をローディングした時、システムコントロールはBASICインタープリタに移されますが、MONコマンドによってシステムコントロールをMONITORの方へ移すことができます。

MONITOR MZ-1Z001Mは、M（メモリ・コレクション）、D（メモリ・ダンプ）、J（ジャンプ）、S（セーブ）、V（ベリファイ）、L（ロード）の6個のコマンドによって機械語プログラムの作成、実行、ファイル出力ができます。MONITOR MZ-1Z001Mについては

MONITOR編

を参照してください。

なお、MONITORからBASICへの復帰は次のジャンプ命令によって実行できます。

＊J
J-ADR．$12A0 ………BASICテキストエリアをクリアして復帰する。
(cold start)

＊J
J-ADR．$1300……… BASICテキストエリアをクリアせずに復帰する。
(hot start)

2.3.4 BOOT

書　式　BOOT

機　能　MZ-2000のシステムIPL（initial program loader）を起動します。

解　説　IPLの起動によって新たに、カセットテープファイルあるいはフロッピーディスケットからシステムプログラムがローディングされます。従って当然、現在システム内にあるBASICインタープリタ、MONITOR、ユーザプログラム等は消去されることになります。

BOOTコマンドの実行は、MZ-2000の電源スイッチをONにした時、あるいはIPLリセットスイッチを押した時と同様の機能があるわけです。

2.4　デファイナブルファンクションキー・リストコマンド

2.4.1　KLIST

書　式　KLIST

機　能　デファイナブルファンクションキーの定義状態を調べるため、CRTディスプレイ上へストリング定義のリスト表示を行います。

解　説　図2－3はKLIST コマンドの実行例を示したもので、10個のファンクションキーの全てが ~~全て~~コマンド等に定義されているのがわかります。

図　2―3

BASIC MZ-1Z001の
ステートメント

Chapter
3

この章は、BASICインタープリタMZ-1Z001の持つすべてのステートメントを説明します。

各ステートメント解説の形式

書　式　ステートメントの正しい一般形を示します。
ここで、プログラマが指定すべき項目は、イタリック体で示しています。
〈 〉内に示した項目は、省略可能であるか任意に繰り返し項目を連ねることができることを示しています。
項目のセパレータ（コンマ、セミコロンなど）は正しく決められた位置に置かなければなりません。

機　能　ステートメントの機能を述べます。

解　説　ステートメントの使い方を詳しく説明します。

例　ステートメントの使い方の例を示します。

3.1 代入文

3.1.1 LET

書　式　<LET>　v=e

v…………数値変数または数値配列要素、あるいはストリング変数またはストリング配列要素（v：variable）

e…………数値定数、数値変数、数値配列要素、数値関数またはそれらの算術式、あるいはストリング定数、ストリング変数、ストリング配列要素またはそれらの結合式（e：expression）

機　能　eで表わされるデータを、vで表わされる変数または配列要素に代入します。

解　説　eの値と、それを代入するvの型は、同じデータ型（数値、ストリング）でなければなりません。LETは、省略することができます。省略した場合、

A=A+3

というような文を、数式の表現として見たとすると"等式"ではあり得ないものです。これは、関係式ではなく、変数Aの値に3を加算して、その結果を新たに変数Aに代入させる文だということを間違わないようにしてください。

例　次の各代入文はいずれも正しい使い方です。

```
10  N=32
10  A$=" Give me file name "
10  LET  A$=A$+CHR$(255)
10  R(D)=SIN(TH*π/180+C)+SIN(2*TH*π/180+C)
I=1  …………ダイレクトモードでの代入例
```

次の各文は誤りです。

```
20  D$=A+B…………左辺がストリング変数であるのに右辺が数式となっている。
20  LOG(LK)=LK+1………左辺が数値変数または数値配列要素でない。
```

3.2 入出力文

一般に入出力文といえば、キーボード、CRTディスプレイ、サウンド、入出力ターミナル、ラインプリンタそして外部ファイルとのやりとりを制御するすべての文が含まれます。しかし、このセクションでは、そのうち基本的な入出力文（PRINT文、INPUT文、GET文、READ～DATA～RESTORE文）を解説し、その他の入出力文は、以降のセクションで、データファイルコントロール文、音楽コントロール文、グラフィックコントロール文、入出力ポートコントロール文にそれぞれグループ分けして解説されています。

3.2.1 PRINT

書　式　PRINT　$\langle e_1\ d_1\ e_2\ d_2 \cdots\cdots\cdots\cdots e_n\ d_n \rangle$

?　$\langle e_1\ d_1\ e_2\ d_2 \cdots\cdots\cdots\cdots e_n\ d_n \rangle$

e_i …………出力データまたはタビュレーション関数

d_i …………セパレータ

機　能　オペランドの出力並びに指定されたデータ（数値データ、ストリングデータ）をセパレータの機能、タビュレーション関数の機能等に従って並びの順にCRTディスプレイに表示します。

解　説　データの出力ポインタはカーソルであり、PRINT文を実行する直前のカーソル位置から表示が実行され、カーソルポインタも移動します。データの出力形式は、セパレータ、TAB関数、SPACE$関数、ストリングデータ中のカーソルコントロールコードによって決まります。

出力並びを省略すると、改行のみが実行されます。

キー入力を簡便にするため、２番目の書式に示したように"？"記号を"PRINT"のかわりに使用することができます。

数値データの出力は、実数形式または、指数形式によって表示されます。

実数形式の出力は、一般に、0.00000001以上10^8未満のデータについて行われ、次の形式で表示されます。

（空白 / －）　X－－－X．X－－－X

（１未満のデータの時は１の位を桁に含めない）

即ち、Xは０～９の数字で８桁以内であり、小数の場合は、数字と数字の間のいずれかに小数点が置かれます。データの先頭に置かれる符号は正ならば空白、負なら－（minus）が表示されます。

指数形式の出力は、実数形式で表示できる範囲を越えた場合行なわれ、次の形式で表示されます。

（空白 / －）．X－－－－－－X　E（＋ / －）YY

即ち、データの先頭に符号がつき、正ならば空白、負ならば－(minus)が表示されます。

仮数部（．X－－－－－X）は小数点以下８桁以内であり、最大の桁と最小の桁は０以外の数字になります。

指数部はE（exponential）表示に続いて、指数の符号（＋または－）が表示され、指数が２桁の数の数YY（Yは０～９の数字）によって表示されます。

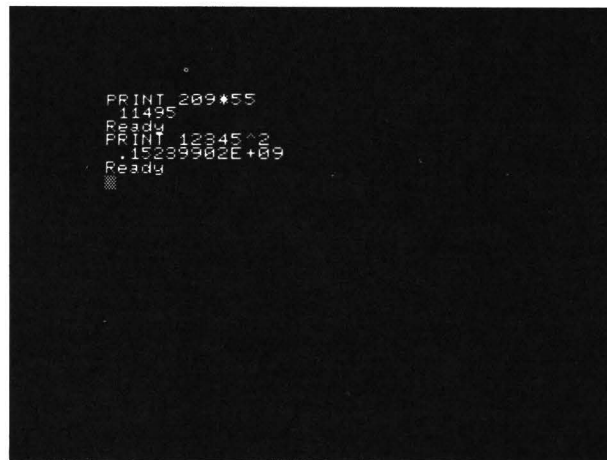



図　3—1

ストリングデータの出力は、カーソル位置から続けて実行されます。

ストリングデータが、たとえばカーソルコントロール等のPRINT文の制御用コードである場合は、それに従ったコントロールが実行されます。即ち、アスキーコード0～29までが問題となるので、それぞれの場合、PRINTの出力並びのデータとした時、どのようなコントロールが行なわれるかをまとめて示します。

PRINT CHR$（0）	何も実行しません。
PRINT CHR$（1）	カーソルが1行分下へ移動します。
PRINT CHR$（2）	カーソルが1行分上へ移動します。
PRINT CHR$（3）	カーソルが1字分右へ移動します。
PRINT CHR$（4）	カーソルが1字分左へ移動します。
PRINT CHR$（5）	カーソルがCRTディスプレイの左上隅へ移動します。
PRINT CHR$（6）	画面をクリアし、カーソルがCRTディスプレイの左上隅へ移動。
PRINT CHR$（7）	カーソルの左側にある1文字を削除し、カーソルも1文字分左へ移動。
PRINT CHR$（8）	カーソルの位置に1文字分の空白をあけます。
PRINT CHR$（9）	キー入力モードについてグラフィックモード⟷ノーマルモードを切り換えます。
PRINT CHR$（10）	キー入力モードについてシフトロックモード⟷ノーマルモードを切り換えます。
PRINT CHR$（11）	何も実行しません。
PRINT CHR$（12）	キー入力モードについてカナモード⟷ノーマルモードを切り換えます。
PRINT CHR$（13）	何も実行しません。
PRINT CHR$（14）	シフトロックモード、グラフィック入力モード、カナモードを全て解除。
PRINT CHR$（15）	キーボードのキー入力モードについてカナモードを解除します。
PRINT CHR$（16～29）	スペースを1個表示します。

上記命令を実行する場合、この命令のあとにセミコロンをつけると上記コントロール内容のみを実行します。

〔例〕

70 PRINT CHR$ (2);

同じく、コンマをつけると上記コントロール実行後、10文字分のタビュレーションを行ないます。

〔例〕

80 PRINT CHR$ (3),

セミコロンまたはコンマをつけないでこの命令を実行する場合は、上記コントロール実行後改行されます。

〔例〕

90 PRINT CHR$ (6)

〔注意〕

キー入力モードコントロールコードCHR$(9)、CHR$(10)、CHR$(12)では、それぞれのキー入力モードとノーマルモードを切り換える、とあります。これは、モード切り換えはフリップ・フロップ動作なので、1回コードを与える毎にキー入力モードが切り換わることになるという意味です。たとえば、現在ノーマルモード（またはカナモードまたはシフトロックモード）のときPRINT CHR$(9)を実行すればキー入力モードはグラフィックモードとなるし、現在グラフィックモードのとき実行すればノーマルモードへ戻ることになります。また、キャリッジリターンを行うとキー入力モードは常にノーマルモードへ戻りますから注意が必要です。

たとえば、INPUT文の実行の前にキー入力モードをカナ入力モードとするには

100 PRINT CHR$(14);CHR$(12);: INPUT A$

とします。セミコロンを忘れるとキャリッジリターンが実行されカナ入力モードはたちまち解除されてしまいます。

複数個の出力並びは、2種類のそれぞれ機能の異るセパレータによって区切ります。

; (セミコロン) …………出力データを続けて表示します。

, (コンマ) ………………出力データをタビュレーションをつけた位置に表示します。即ち、左のデータの先頭位置から10文字目を右のデータの先頭位置とします。ただし、左のデータが10文字以上ある場合は、オーバレイしない位置まで、10文字単位のタビュレーションを行ないます。

TAB関数（p.95参照）はPRINT文中で、タビュレーションコントロールファンクションとして用いられ、任意のタビュレーションを与えることが可能です。(タビュレーションは増加する形でしか与えられません)

3.2.2 INPUT

書　式　INPUT　〈x\$ ；〉 v_1 〈 , v_2 , ……, v_n 〉

x\$…………ストリングデータ：データ入力メッセージ

v_i …………数値変数または数値配列要素、あるいはストリング変数またはストリング配列要素

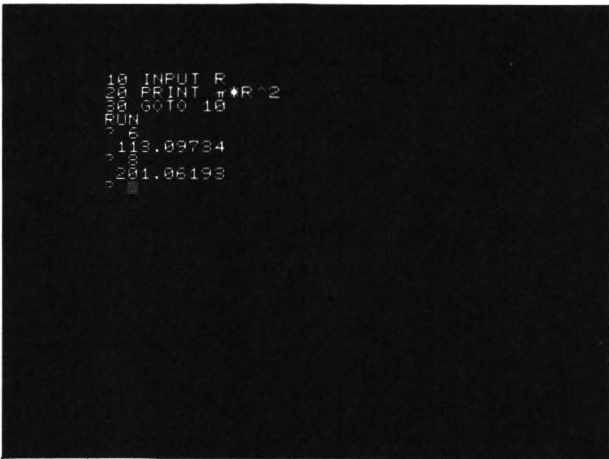
機　能　プログラムの実行中に、キーボードからデータ（数値定数、ストリング定数）が入力されるのを待ち、入力並びに指定された変数または配列要素にデータを順次代入します。

解　説　INPUT文が実行されるとCRTディスプレイに"？"が表示され、つづけてカーソル点滅によってデータの入力待ちの状態にあることを示します。"ストリングメッセージ"を用いている場合は、はじめにそのメッセージを表示し、続けてカーソルを表示します。"ストリングメッセージ"と入力並びのセパレータは"；"を用います。

入力並びが複数個ある場合、入力データをコンマ"，"によって区切ってキー入力することができます。データ入力はCRキーによって実行されます。

入力データが入力並びより少い場合は、次の行に"？"を表示して更にデータが入力されるのを待ちます。

入力データが入力並びより多い場合、余ったデータは無視されます。入力データと入力並びのデータの型式は同一でなければなりません。ストリング定数は、クォーテーションマークを用いずに入力できますが、その場合はストリングデータの前後に与えられたスペースは無視されます。ストリングの前後にスペースを与えたり、コンマを与える場合は全体をクォーテーションマークで囲みます。

図　3−2

3.2.3 GET

書　式　GET　v

v…………数値変数または数値配列要素、あるいはストリング変数またはストリング配列要素

機　能　プログラムの実行時に、押されているキーを調べてそのデータを変数または配列要素に代入します。

解　説　GET文が実行された時、キーが押されていたら、そのデータを変数または配列要素に代入します。キーが押されていない場合、数値変数または数値配列要素には0が代入され、ストリング変数またはストリング配列要素には" "（Null）が代入されます。

GET文によって入力されるデータはどれか1つのキーデータであり、数値データであれば、0～9のうちいずれかの整数であり、ストリングデータであれば、1文字が入力されます。2個以上のキーを押していても、優先度の最も高いキーが有効となります。また、数値変数または数値配列要素に対しては、数字のキー以外を押していても無視されます。

3.2.4 READ～DATA

書　式　READ　v_1〈, v_2, …………, v_m〉

……………………………………

DATA　d_1〈, d_2, …………, d_n〉

v_i………入力並び：数値変数または数値配列要素、あるいはストリング変数またはストリング配列要素

d_i………データ並び：数値データまたはストリングデータ

機　能　DATA文に記述したデータ並びを読み、変数または配列要素に代入します。

解　説　READ文とDATA文はいつも対にして用います。

DATA文は、1文あるいは複数文で、データ並び（データテーブル）を構成し、READ文は、そのデータを順番に1つ1つ読み出して、入力並びの1つ1つの変数または配列要素に代入します。従って、データ並びのデータと、対応する入力並びの変数または配列要素のデータの型は一致しなければなりません。一致しないとError 4（Data mismatch）が発生します。

READ文の実行中にDATA文のデータ並びを越えた場合は、Error 24（Out of DATA）が発生します。

READ文で、データ並びの途中のデータまで読んだあと、次のREAD文を実行すると、その次のデータからひき続いて読み出されます。

RESTORE文を用いると、次に実行するREAD文で読み出すデータ並びの位置を決めることができます。

3.2.5 RESTORE

書　式　RESTORE　< *line number* >

機　能　READ文で読み出すデータ並びを、データ並びの先頭、あるいは指定した行番号のデータ並びに設定します。

解　説　オペランドを記述しないかあるいは、0を記述すると、データ並びの先頭、即ち、DATA文のうち最も小さい行番号をもつものの先頭へデータ読み出しポインタがリストアされます。
オペランドに<行番号>を指定すると、その行のDATA文、または、その以後で最も小さい行番号をもつDATA文の先頭へデータ読み出しポインタがリストアされます。

3.3 ループ文

3.3.1 FOR～NEXT

書　式　FOR　*cv*=*iv* TO *fv* <STEP *sv*>

　　　　　　……………………

NEXT <*cv*>

cv…………コントロール変数(contol variable)：数値変数または数値配列要素

iv………… 初期値 (initial value)：数値データ

fv………… 終値 (final value)：数値データ

sv………… 増分 (step value)：数値データ

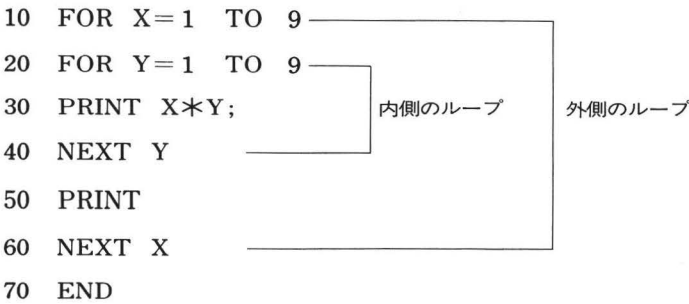
機　能　あるルーチンの繰り返し実行を行います。

解　説　変数*cv*は、FOR～NEXT 文が構成する繰り返しブロック (ループ) の制御変数であり、最初初期値*iv*が代入されます。ルーチンの終わりでNEXT文が実行され、STEP *sv*で与えられた増分が制御変数に加算され、この値が終値*fv*を越えない値であれば、FOR文の次の実行文へ戻り、繰り返し処理が続けられます。増分*sv*は、通常は正の増分を与えますが、負の値を用いることもできます。このとき最終値は初期値より小さく設定します。ループは、制御変数の値が最終値より小さくなるまで実行されます。STEP *sv*を省略すると、増分＝1で実行されます。

多重ループ

FOR～NEXTループは多重に重ねることができます。ループの重ね合わせは、必ず入れ子型(ネスト構造) となるようにし、内側のループは、外側のループに完全に含まれなければなりません。またそれぞれのループは別々の制御変数を使わなければなりません。

ネストの深さは最大16 (即ち16重のループ) です。ネスティングが16重を越すとError 11(FOR nesting) が発生します。

多重ループが同じ場所で終わる場合は、1つのNEXT文にまとめてしまうことができ、その時はNEXT 文のオペランドには制御変数名をコンマで区切って、内側のループの制御変数名から順に記述します。

NEXT 文が最も近い FOR 文に対応する場合は、その制御変数名を省略することができます。もし対応するFOR文のないNEXT文があると、そこで Error 13 (NEXT-no FOR) が発生します。

FOR～NEXTのループ中ではCLR文は使えません。

例　プログラム

```
10  FOR  X=1  TO  9  ──────────────┐
20  FOR  Y=1  TO  9  ───┐          │
30  PRINT  X*Y;         │内側のループ │外側のループ
40  NEXT  Y  ───────────┘          │
50  PRINT                          │
60  NEXT  X  ──────────────────────┘
70  END
```

この場合ネスティングの深さは2です。図3－3は、2重ループを用いて九九を表示させた例です。

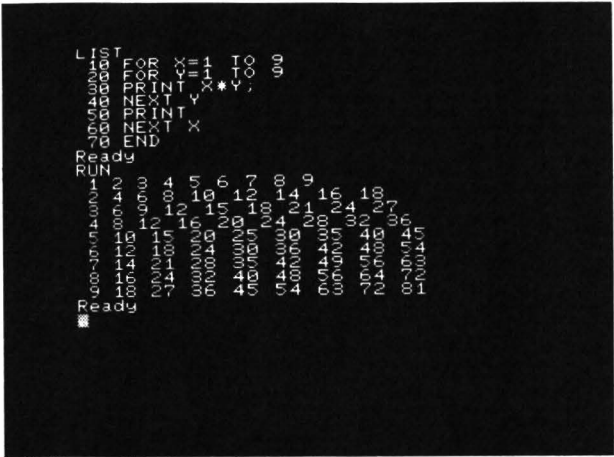
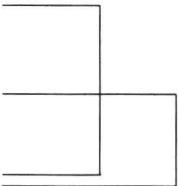



図 3－3 2重ループの例

次は正しくない例です。

プログラム

```
  ⋮
100  FV=30
110  FOR N=0 TO FV STEP3
120  PRINT A$(N)
130  FOR I=1 TO 5
140  PRINT A (N, I)
150  NEXT N, I
  ⋮
```

ネスト構造ができていない

3.4 分岐文

3.4.1 GOTO

書　式　GOTO *lr*

lr…………参照行番号 (reference line number)

機　能　プログラム実行を、*lr*で指定された行番号の文へ無条件に移します。

解　説　*lr*の行の文が実行文であれば、その文およびそれに引き続く文が実行されますが、非実行文であれば、引き続く最初の実行文までスキップします。

(*lr*は、*lr*で指定する行の定義行番号 *ld*を直接指定しなくてはなりません。)

3.4.2 GOSUB～RETURN

書　式　GOSUB *lr*

………………

RETURN

lr…………参照行番号

機　能　プログラム実行を、*lr*で指定された行番号のサブルーチンへ無条件に移し、RETURN文によって、GOSUB文の次の文へ戻ります。

解　説　サブルーチンは、プログラム中である特定の問題を処理する目的で１つのルーチンとしてまとめられたものです。プログラム中で何度も実行する必要のある部分をサブルーチン化することによって、テキストを縮めることができるだけでなく、プログラムの構造自体を組織化することができます。サブルーチンは、メインプログラムあるいは他のサブルーチンから参照されますが、サブルーチンの多重化（ネスティング）は、最大16まで許されます。

サブルーチンの分岐とはGOSUB文の実行自体が、その定義となっています。従ってサブルーチンの先頭や終わりといった特別の場所が必ず存在する訳ではなく、GOSUB文によるプログラムの分岐と、対応する同じレベルのRETURN文によって復帰がそれぞれ実行されます。たとえば、１つのサブルーチンに対して複数のちがった参照行番号による呼び出しが行われたり、複数のRETURN文によってサブルーチンの終了が決められることがありますが、もちろん、それぞれの場合には、１つのGOSUB文の実行に対して、１つのRETURN文の実行が対応しなくてはなりません。

3.4.3 IF～THEN

書　式　IF　*e*　THEN　*lr*

IF　*e*　THEN　*statement*

e…………関係式 (relational exression) または論理式 (logical expression)

lr…………参照行番号 (reference line number)

機　能　関係式または論理式 *e* によって記述される条件を判断し、条件付きの分岐を実行します。

解　説　関係式または論理式が真の場合（条件が成立し、式の値が－1となる場合）は、THEN以下が実行されます。THENに続けて参照行番号 *lr* を記述すると、その行へのジャンプが実行されます。

THEN以下には、続けて別のステートメントを記述することが可能であり、IF　THEN　IF～といった構文を作成することもできます。

関係式または論理式が偽の場合（条件が成立せず、式の値が0となる場合）は、次の行番号へプログラム実行が移ります。従って、IF～THEN 文の後にマルチステートメントが記述されていても、この場合は無視されることになります。

IF～THEN 文の働きを論理図式で表わすと次のようになります。

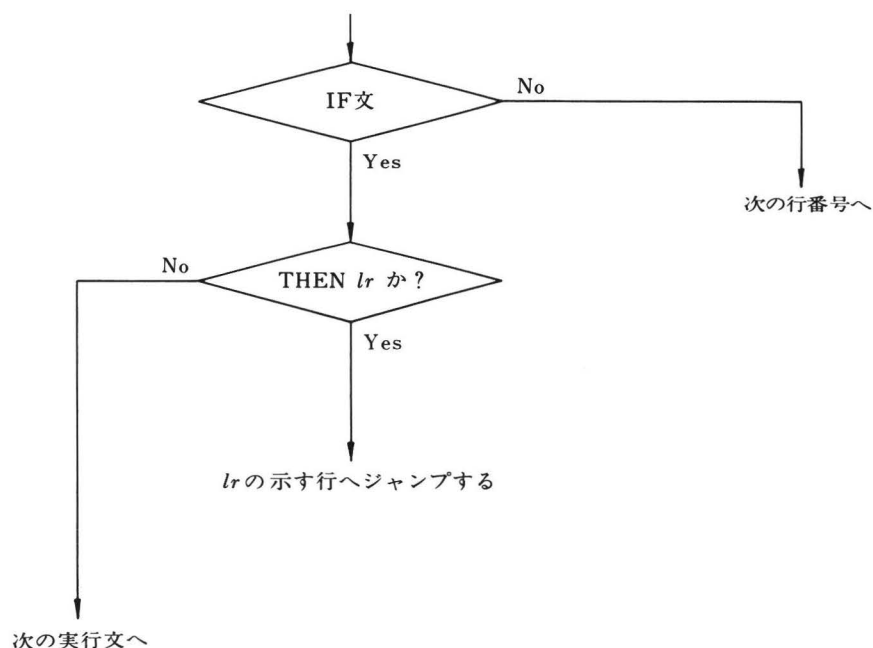



図3—4 IF～THEN文の機能

数値データを比較する場合の注意

BASIC　MZ-1Z001で扱う数値は、BASIC内部では2進浮動小数点形式で記述され、演算、比較処理が実行されます。2進数表現の場合、数値は必ずしも正確でないため、実数演算、比較等の処理では、誤差の発生あるいは、外部表現（PRINT文などによって示される10進表現）とのくいちがいが発生し得るということに注意する必要があります。たとえば、ある演算の結果の値が、数学的には整数値となる場合でも、演算の途中で整数以外のデータを扱っている場合には、内部表現では必ずしも整数値が得られるとは限りません。

例

数値データの内部表現と外部表現の相違を簡単な例によって見てみることにします。たとえば、1を10で割って、次に10倍するという演算を行うと、結果の表示は"1"となりますが内部表現が、"整数の1"でないためIF文の比較で両者が一致しないということが起きます。

プログラム

```
10 A＝1/10＊10
20 IF A＝1 THEN PRINT "TRUE"：GOTO 40
30 PRINT "FALSE"
40 PRINT "A＝"; A
50 END
```

実行

```
RUN
FALSE
A＝  1
```

この実行結果はA＝1と表示されながら比較が"FALSE"であり、一見不合理ですが、数学的な数のイメージと計算機が扱う固定長数値データとは同じものではあり得ないこと、計算機で扱う数値データも内部表現と外部表現とが異ることがあることを示しています。

IF文での数値データの比較は数値の内部表現によっています。そこで比較文を、外部表現のレベルにしてやることによって上記の不合理さを除くことができます。内部表現を外部表現へ変える場合は、有効桁数8桁内の数値へ置き換える操作が行われるので、次のIF文、

```
IF ABS(A－X)＜ .1E－8 THEN………
```

によって数値データAとXは、$1/10^8$ の精度、従って数値の外部表現のレベルで比較されることになります。

練 習

1） 行番号20を上記のABS関数を用いた比較文に変更し、精度を変えてみて、Aの値と整数1との違いがどの程度のものか調べよ。

2） 行番号10中の演算を、順序を変えて

```
10 A＝1＊10/10
```

とした時はどうであるか調べよ。

演算子	応用例	説明
＝	IF A＝X THEN…	変数AとXの数値が等しいならば、THEN以降の命令を実行します。
	IF A＄＝"XYZ" THEN…	ストリング変数A＄の内容がストリングXYZであれば、THEN以降の命令を実行します。
＞	IF A＞X THEN…	変数AがXより大きいならば、THEN以降の命令を実行します。
＜	IF A＜X THEN…	変数AがXより小さいならば、THEN以降の命令を実行します。
＜＞or＞＜	IF A＜＞X THEN…	変数AとXの数値が等しくないならば、THEN以降の命令を実行します。
＞＝or＝＞	IF A＞＝X THEN…	変数AがXより大きいか等しいならば、THEN以降の命令を実行します。
＜＝or＝＜	IF A＜＝X THEN…	変数AがXより小さいか等しいならば、THEN以降の命令を実行します。
＊	IF(A＞X)＊(B＞Y) THEN…	変数AがXより大きく、かつ変数BがYより大きいならば、THEN以降の命令を実行します。
＋	IF(A＞X)＋(B＞Y) THEN…	変数AがXより大きいか、または変数BがYより大きいならば、THEN以降の命令を実行します。

3.4.4 IF～GOTO

書式 IF *e* GOTO *lr*

e…………関係式または論理式

lr…………参照行番号

機能 関係式または論理式*e*によって記述される条件を判断し、その結果に従って分岐を行います。

解説 IF～THENと同様に条件付き分岐を行う文ですが、条件が成立する場合参照行番号*lr*の行へのジャンプを実行します。

条件が成立しない場合は、次の行へプログラム実行が移ります。IF～GOTO *lr* とIF～THEN *lr* は同じ働きがあります。IF～GOTO文以降には、マルチステートメントを置いてもプログラム実行にとって無意味となるので注意が必要です。

3.4.5 IF～GOSUB

書式 IF *e* GOSUB *lr*

e…………関係式または論理式

lr…………参照行番号

機能 関係式または論理式*e*によって記述される条件を判断し、その結果に従って、サブルーチンへの分岐を行います。

解説 IF～THEN文と同様の条件付き分岐を行う文ですが、条件が成立する場合、参照行番号*lr*で示されるサブルーチンを呼び出します。RETURN文によるサブルーチンからの復帰は、IF～GOSUB文の次の実行文（従ってマルチステートメントであれば、IF～GOSUB文に引きつづく文）へ行われます。

3.4.6 ON～GOTO

書　式　ON　e　GOTO　lr_1 <, lr_2, lr_3, ………, lr_n>

e ………数値変数、数値配列要素またはそれらの算術式

lr_i ……参照行番号：分岐先の並び

機　能　eの値に従って、いくつかの指定された行番号の行への多岐分岐を行ないます。

解　説　ON～GOTO文による分岐先は、GOTOにつづいて、いくつかの参照行番号を記述することにより（セパレータは"，"）指定しますが、分岐先の数は1行に記述できるなら幾つでもかまいません。これら第1番目から第n番目までの分岐先に対して、整数1、2…………nが対応し、eの値が、整数値mであれば、第m番目の分岐先へプログラム実行が移されます。mが1～n以外であれば、ON～GOTO文の次の実行文へ移ります。

eの値（内部表現の値）が整数でない場合、小数部分を切り捨てた整数値によって分岐先が決められます。従って数値の内部表現に対する注意（IF～THEN文中の注意を参照して下さい）が必要です。たとえば、Aの値の内部表現が、1.99999999…………という値であったとしたら、ON文で扱われる値は、この整数部の値、即ち1となります。

3.4.7 ON～GOSUB

書　式　ON　e　GOSUB　lr_1 <　lr_2　lr_3　………　lr_n >

e …………数値変数、数値配列要素またはそれらの算術式

lr_i ………参照行番号

機　能　eの値に従って、いくつかの指定された行番号のサブルーチンを呼び出します。

解　説　ON～GOSUB文の基本的動作は、ON～GOTO文と同様ですが、分岐先は、全てサブルーチンとなります。サブルーチンからの復帰は、ON～GOSUB文の次の実行文へ行われます。従って、どのサブルーチンへ分岐しても、RETURN文による復帰は同一になります。

3.5 定義文

3.5.1 DIM

書　式

DIM a_1 (i_1) < , $a_2(i_2)$, ………… , $a_m(i_m)$ >

DIM b_1 (i_1, j_1) < , b_2 (i_2, j_2) , ………$b_n(i_n, j_n)$>

a_i …………1次元配列名（リスト）

b_i …………2次元配列名（テーブル）

i_m、i_n、j_n…………ディメンジョン

機　能

1次元配列または2次元配列のディメンジョンを宣言し、必要なメモリ領域を確保します。(DIM：dimension)

解　説

1次元配列または2次元配列（数値配列、ストリング配列）を使用するには、DIM文によって、あらかじめ配列の名前と大きさを宣言しておかなければなりません。配列名は大文字2文字が有効です。配列の添字は、0から最大255まで使えますが、メモリの使用状況によって、制限されることがあります。

1つのDIM文で複数の配列の配列宣言を行うには、オペランドを" , "で区切って記述します。

1次元配列（リスト）の配列宣言

DIM　A(20) ……………………1次元数値配列A(　)について、配列要素をA(0)からA(20)まで21個用意します。

DIM　ST$(99) ……………1次元ストリング配列ST$(　)について、配列要素をST$(0)からST$(99)まで100個用意します。

2次元配列（テーブル）の配列宣言

DIM　N1(5 ,11) …………2次元数値配列N1(,)について、配列要素をN1(0 ,0)からN(5 . 11) まで、6×12=72個用意します。

DIM　S$(11 , 30) …………2次元ストリング配列S$(,)について配列要素をS$(0 ,0) からS$(11 , 30)まで、12×31=372個用意します。

DIM文を実行した時点では、宣言した配列の、すべての要素の値を0または" "(空)にします。

すでに配列宣言を行っている配列に対して、より大きい配列を宣言すると、Error 7 (Dimension overflow)が発生します。

CLR文が実行されると、配列宣言は全て無効になります。

3.5.2 DEF FN

書　式　DEF　FNf (x) $=e$

f …………利用者関数名：アルファベットの大文字１文字A～Z

x …………変数名

e …………数値定数、数値変数、数値配列要素、数値関数、定義済みの他の利用者関数などによる算術式

機　能　利用者関数FNf(x)を定義します。

解　説　任意の１変数関数を定義することができます。関数名は､FNにつづけてアルファベット１文字で指定し、(　)内に変数名を指定します。従って同時に定義できる利用者関数は最大26となります。またネスト構造による利用者関数定義も可能であり、ある利用者関数の定義式中にすでに定義されている他の利用者関数を用いることができます。利用者関数定義のネスティングは最大６重まで可能です。６重を越すと、Error 12 (function nesting) が発生します。１つのDEF文では、１つの利用者関数しか定義できません。組み込み関数 (built in functions) の他の標準関数 (三角関数、逆三角関数、双曲線関数) の定義の例は、巻末の表A. 4 に示されています。

CLR文が実行されると、利用者関数はすべて未定義となります。

例　図３－５は、三角関数によって２つの関数FNX(T)とFNY(T)を定義して繰り返し曲線を描かせた例です。

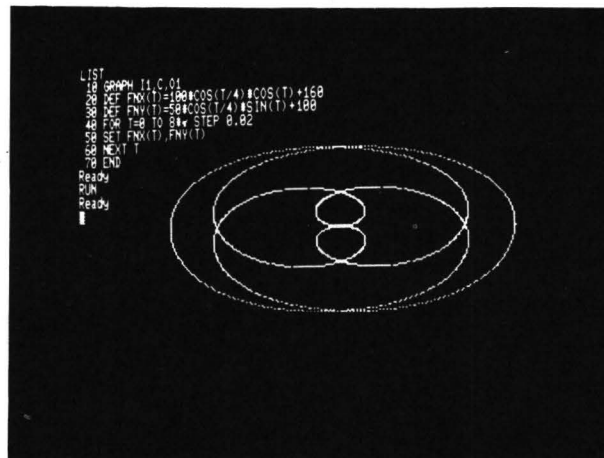



図　3—5

〔正例〕

10 DEF FNA(X)＝TAN(X－π/6)

20 DEF FNB(X)＝FNA(X)/C＋X………ネスティングを用いた関数定義

〔誤例〕

10 DEF FNK(X)＝SIN(X/3＋π/4) , FNL(X)＝EXP(－X^2/K)

………DEF文のオペランドに２つの関数定義式を置くことはできない。

3.5.3 DEF KEY

書　式　DEF KEY (*k*) ＝*s*

k…………デファイナブルファンクションキーの番号：1～10

s…………キャラクタストリング

機　能　デファイナブルファンクションキーを定義します。(DEF KEY : define key)

解　説　10個のファンクションキーのいずれかに対して、その機能を定義します。キー番号*k*に1～10のいずれかを指定し、定義式の右辺に、ファンクションを示す、ストリングやコマンド等をそのまま記述します。ファンクションに実行機能を持たせるため、キャリッジリターン機能を持たせることも可能であり、SFT LOCK と GRPH キーの両方を同時に押すと記号"↴"が入力され、これがファンクションキーの使用時にキャリッジリターン機能を持ちます。

同じファンクションキーについて、新たにDEF KEY文が実行されると前の定義は無効となります。デファイナブルファンクションキーに、マルチコマンドを設定するには、セパレータ"！"によってコマンドの区切りとします（"："は使えません)。

DEF KEY文のあとに別のステートメントを置くことはできません。

```
LIST
 10 DEF KEY(1)=LIST↴
 30 DEF KEY(3)=RUN↴
 50 DEF KEY(5)=SHARP Personal Computer M
Z-2000
 70 DEF KEY(7)=CONSOLE
 80 DEF KEY(8)=GRAPH
 90 DEF KEY(9)=?CHR$(6);!KLIST
Ready
RUN
Ready
KLIST

1 LIST↴
2
3 RUN↴
4
5 SHARP Personal Computer MZ-2000
6
7 CONSOLE
8 GRAPH
9 ?CHR$(6);:KLIST
10
Ready
```

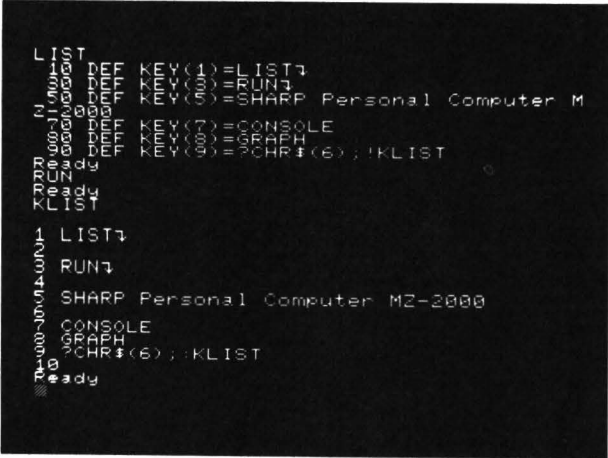



図 3—6

3.6　注釈文とコントロール文

3.6.1　REM

書　式　REM *r*

r…………注釈文

機　能　プログラムリスト上の注釈文です。(REM : remark)

解　説　REM文はプログラムリストを見易くするための注釈文の役割を果すもので、非実行文(nonexecutable statement)　です。プログラムの実行は、REM文をスキップして、次の実行文へ移されます。

3.6.2 STOP

書　式 STOP

機　能 プログラムの実行を一時停止し、コントロールをコマンドレベルへ戻します。

解　説 STOP文を実行するとシステムは、

```
*STOP IN 1050
Ready
```

というように、そのSTOP文の置かれている定義行番号(上例では1050)をCRTディスプレイに表示してプログラムの実行を停止します。ストップした状態で、各データの内容を調べたりした後に、CONT コマンドによってプログラムの実行を継続させることができます。ただし、その場合、プログラムの変更等のエディション操作を行うと、CONTコマンドは無効となります。(2.3.2節のCONTコマンドを参照のこと) STOP 文は、マルチステートメント中のどの位置にも置くことができます。

STOP文はEND文と異なり、ファイルのクローズを実行しません。

3.6.3 END

書　式 END

機　能 プログラムの実行を終え、コントロールをコマンドレベルへ戻します。

解　説 END文を実行するとシステムは、

```
Ready
```

を表示して、BASICインタープリタのコマンドレベルが"Ready"状態となったことを示します。その際、プログラム実行中で、オープンされたままのファイルがあれば、それをクローズします。

END文のあとに、実行文が続く場合は、前記STOP文と同様に、CONTコマンドによって、プログラムの実行を継続させることが可能です。

プログラムテキストの最後の実行文でプログラム実行を終える場合は、END文は必要ありません。

3.6.4 CLR

書　式 CLR

機　能 変数および配列をすべて未定義状態にします。(CLR：clear)

解　説 変数については、数値変数の値をすべて0に、ストリング変数の内容をすべて" " (空) にします。

配列は、すべての配列を未定義状態に、即ち、既に実行されているDIM文をすべて無効とします。従って、CLR文の実行後に、配列を使用するには、新たにDIM文によって必要とする配列のディメンジョン定義を行わなければなりません。

なお、既に実行されているDEF FN文もすべて無効とします。従って、CLR文の実行後に、利用者関数を使用するには、新たにDEF FN文によって必要とする利用者関数の定義を行わなければなりません。

FOR、NEXTの中、並びに、サブルーチンの中にCLR命令を置くことはできません。

3.6.5 CURSOR

書　式　CURSOR *x*，*y*

x…………X-座標：数値データ

y…………Y-座標：数値データ

機　能　CRTディスプレイの座標位置を指定して、そこへカーソル・ポインタを移動させます。

解　説　PRINT文、INPUT 文によるメッセージの表示は、カーソル・ポインタ位置から実行され、それらの各文を実行するにつれて、カーソルポインタも移動して行きますが、CURSOR文を使うことによって、CRTディスプレイ上の任意の位置にカーソル・ポインタを移すことができます。

座標位置は、X座標：*x*、Y座標：*y*の順に、オペランドに記述します。各値の範囲は、80―キャラクタ表示モード、40―キャラクタ表示モードでそれぞれ次のようになっています。

■80―キャラクタ表示モード

X-座標：0 ～79

Y-座標：0 ～24

■40―キャラクタ表示モード

X-座標：0 ～39

Y-座標：0 ～24

もし、*x*、*y*の値が整数でない場合は、小数点以下は切り捨てられます。

CONSOLE S文によってスローリングエリアが限られている場合でも、そのエリア外の任意のY-座標を指定することができ、PRINT文によってメッセージを表示させることができます。

3.6.6 CSRH

書　式　CSRH

機　能　現在のカーソルの水平位置を与えるシステム変数（CSRH：cursor，horizontal）

解　説　カーソル位置は、CURSOR文、PRINT文、INPUT文などの実行によって移動しますが、システム変数CSRHは、X軸上のカーソル位置を示します。CSRHのとり得る値は、各キャラクタ表示モードでそれぞれ次のようになります。

80キャラクタ表示モード：$0 \leqq CSRH \leqq 79$

40キャラクタ表示モード：$0 \leqq CSRH \leqq 39$

3.6.7 CSRV

書　式　CSRV

機　能　現在のカーソルの垂直位置を与えるシステム変数（CSRV：cursor，vertical）

解　説　CSRVのとり得る値は、各キャラクタ表示モードで同様です。即ち、

$0 \leqq CSRV \leqq 24$

3.6.8 CONSOLE

書　式　CONSOLE 〈Sy_s , y_e , C$_n$, R , N , GH , GN〉

y_s ………スクローリングエリアの先頭行のＹ－座標

y_e ………スクローリングエリアの終行のＹ－座標

n ………１行のキャラクタ数：80または40

機　能　CRTディスプレイ上のスクローリングエリアの設定、キャラクタ表示モード（80－キャラクタ・モード／40－キャラクタ・モード）の設定、画面全体の白黒反転および解像度の切り替えを行います。

解　説　CONSOLE文のオペランドの記述によって、次の４種類の機能があります。

■スクローリングエリアの設定

CRTディスプレイ上のスクローリングエリアの設定は、次の書式で行います。

CONSOLE　Sy_s, y_e　　　　　　　　(S：scroll)

ここで

$0 \leqq y_s < y_e \leqq 24$　　かつ、　$y_s + 2 \leqq y_e$

でなければなりません。

スクローリングエリアとは、そのエリアの終行から次の行へカーソルを移動させる時、画面が１行上へ繰り上がるエリアであり、BASIC インタープリタの起動時は、第０行から第24行、即ち画面全体がスクローリングエリアになっています。

図３－７の左側の画面は通常の状態であり、LIST コマンドによるリスト表示も画面全体に行われていますが、右側は、第10行から第20行にスクローリングエリアを限った状態であり、同じリスト表示も画面の中央部に限られていることが示されています。

```
 15 PRIM(0,0)=2
 20 GOSUB 1000
 30 GOSUB 2000
 40 END
 1000 I=1 : FOR X=0 TO 9 : FOR Y=0 TO 99
 1010 GOSUB 1500
 1020 PRIM(X,Y)=I:PRINT I;
 1030 NEXT Y,X
 1040 RETURN
 1500 I=I+1 : M=0:N=0
 1510 IF INT(PRIM(M,N)^2+0.00000001)>I T
HEN RETURN
 1520 L=I/PRIM(M,N)
 1530 IF ABS(L-INT(L))<0.00000001 THEN 1
500
 1540 N=N+1
 1550 IF N<100 THEN 1510
 1560 M=M+1:N=0:GOTO 1510
 2000 FOR M=0 TO 9:FOR N=0 TO 99
 2010 PRINT PRIM(M,N);
 2020 NEXT N,M
 2030 RETURN
Ready
```

図　3－7

スクローリングエリアを限定した場合、そのエリア外の画面は、スクロールによって画面から消えたり、CLR/HOME キーのクリア操作、あるいは、PRINT　CHR＄(6)のクリア操作によって消されることがありません。従って、画面上に常に表示させておきたい部分があるような時に、CONSOLE S文によってそのエリアを、スクローリングエリアから除外させてしまうといった使い方ができます。尚、スクローリングエリア外であっても、CURSOR 文によって、カーソルポインタを持って行くことができるので、メッセージをそこに表示することは可能です。

■キャラクタ表示モードの設定

CRTディスプレイ上の１行に表示するキャラクタ数は、次の書式で設定します。

CONSOLE C*n*　　　　　　　　　　(C：character)

n =80　または　*n* =40

BASICインタープリタの起動時は、40キャラクタ／行のモードとなっています。キャラクタ表示を80キャラクタ・モードとすると、各キャラクタは、X軸方向に$\frac{1}{2}$に圧縮されて小さく見えますが、当然１画面にたくさんの情報を表示させることができる訳で、項目数の多い表を作る場合などに使われます。

なお、このCONSOLE C*n*命令を実行すると、グラフィックディスプレイ出力モードページは、全ページともOFFとなります。

図３－８は、40キャラクタモード、80キャラクタモードでのリスト表示を示しています。

```
 40 J$=CHR$(192)+CHR$(192):K$=CHR$(240)+
CHR$(240)
 50 L$=CHR$(12)+CHR$(12):M$=CHR$(63)+CHR
$(63):N$=CHR$(48)+CHR$(48)
 55 O$=CHR$(51)+CHR$(51):P$=CHR$(243)+CH
R$(243)
 60 I1$=H$+N$+O$+O$+M$+B$+A$+H$+H$+P$+H$
+D$+D$+M$+D$+J$+H$+G$+O$+P$+D$+C$+K$+J$
 70 I2$=N$+B$+A$+M$+C$+M$+B$+H$+H$+M$+J$
+D$+M$+D$+D$+M$+A$+C$+K$+D$+B$+D$+I$
 100 GRAPH I1
 110 FOR X=40 TO 260 STEP 30
 120 FOR Y=50 TO 110 STEP 20
 130 POSITION X,Y: PATTERN 16,I1$
 140 NEXT Y,X
 200 GRAPH I2,O2: FOR X=40 TO 260 STEP 3
0: FOR Y=50 TO 110 STEP 20
 230 POSITION X,Y: PATTERN 16,I2$: NEXT
Y,X
 1000 GRAPH O1: GOSUB 2000: GRAPH O2: GO
SUB 2100: GOTO 1000
 2000 MUSIC "R1G0G-G": RETURN
 2100 MUSIC "R1C0#CC": RETURN
Ready
```

図　3－8

■画面全体のノーマル／リバース表示の設定

CRTディスプレイ上のキャラクタ表示、グラフィック表示の白黒をノーマル／リバースのいずれかに設定するには、それぞれ、

CONSOLE R　……リバース表示にする　　　(R：reverse)

CONSOLE N　……ノーマル表示に戻す　　　(N：normal)

を実行します。図３－９の左の例がノーマル表示、右がリバース表示です。

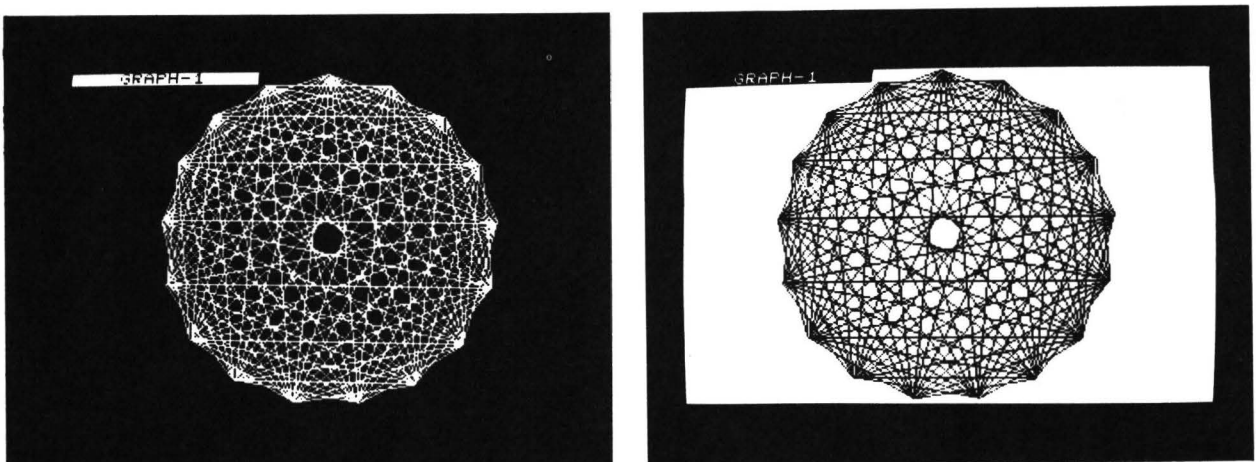



図　3－9

■グラフィック表示の解像度の切り替え

CRTディスプレイ上のグラフィック表示の解像度を640×200ドット／320×200ドットのいずれかに設定するには、それぞれ、

CONSOLE GH……ハイリゾリューションモードにする…………640×200ドット／画面

CONSOLE GN……ノーマルリゾリューションモードにする……320×200ドット／画面

を実行します。

CONSOLE GH（またはGN）を実行後は、グラフィックエリア／ページ1、2、3はすべてクリアされ、グラフィックディスプレイの入出力モードページは、

入力ページモード：ページ1

出力ページモード：全ページともOFF

となり、またポジションポインタは

POSH：0

POSV：0

となります。なお、各キャラクタ設定モード（スクローリングエリア、40／80キャラクタモードおよびノーマル／リバースモード）は、以前設定されたものを持続します。

CONSOLE文のオペランドは、上記の各命令をどの順に記述してもかまいません。またこれらはいずれもモード設定文であり、新たにモード設定されない限り各モードは持続されます。

3.6.9 CHANGE

書　式　CHANGE

機　能　メインキーボードで、アルファベット入力キーの大文字／小文字入力ポジションを逆にします。

解　説　BASICインタープリタの起動時には、アルファベットの小文字はシフトポジション、即ち SHIFT キーを押した状態でキー入力されますが、CHANGE 文を実行すると、大文字／小文字のポジションが逆となり、SHIFT キーを押した状態では大文字が入力されるようになります。この交換は、図3－10でアミで示したA～Zまでの26個のアルファベットキーのみ行われます。

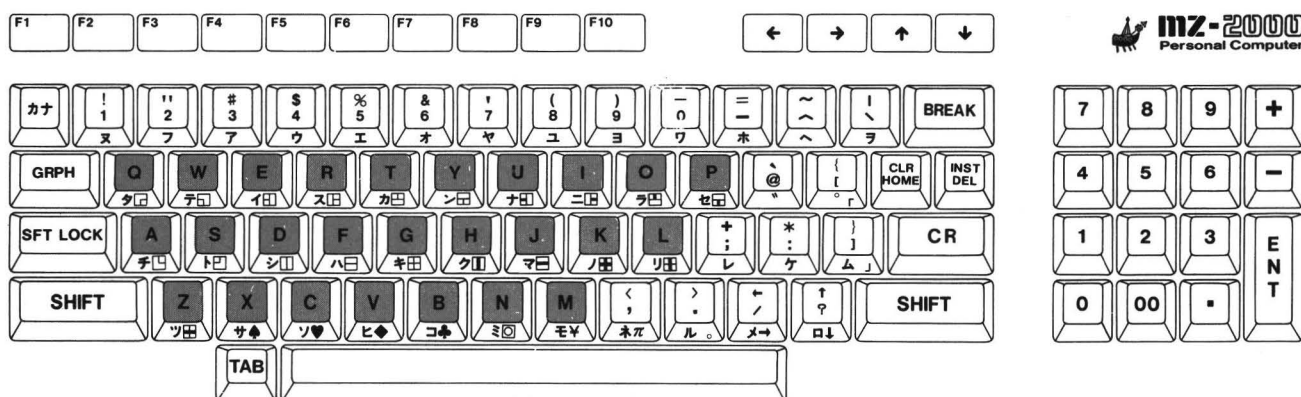



図　3−10

BASICのコマンド、ステートメント、変数名などは、全て大文字で記述しなければならないので、MZ-2000 のメインキーボードでもはじめは大文字がノーマルポジションに置かれている訳ですが、一方、英文によるメッセージの記述の際には逆に、大文字をキー入力する場合の方が少なく、一般のタイプライタと同じように、大文字の方をシフトポジションとした方が使い勝手がよいといえるので、その場合この文によって大文字／小文字を逆にします。

CHANGE 文は、実行するたびにモードが逆になります。

3.6.10 REW

書　式　REW

機　能　カセットテープを巻き戻します。　(REW：rewind)

解　説　REW 文は、カセットテープデッキコントロールの REW キーを押したのと同じ働きがあり、カセットテープの巻き戻しを行います。

REW 文は、カセットテープの巻き戻しを開始したらすぐに次の実行文へコントロールを移します。

カセットテープデッキにカセットテープがない場合は、

　　SET TAPE

の表示が現われ、ウィンドウを開けてカセットテープがセットされるのを待ちます。

カセットテープが既に巻き戻されているときは、続けて次の実行文へコントロールを移します。

3.6.11 FAST

書 式 FAST

機 能 カセットテープの早送りを実行します。 (FAST：fast-forward)

解 説 FAST文は、カセットテープデッキコントロールの FF キーを押したのと同じ働きがあり、カセットテープの早送りを行います。

FAST文も、REW文と同じく、カセットテープの早送りを開始したらすぐに次の実行文へコントロールを移します。

3.6.12 SIZE

書 式 SIZE

機 能 現在のメモリのうちで、BASICプログラムのための空き領域のサイズを示すシステム変数。

解 説 BASICインタープリタの起動時に、インタープリタのバージョン表示に続いてバイトサイズの表示が行われますが、そのサイズは、空きエリアの最大サイズを示しています。たとえば、

38000 Bytes

という表示であれば、38Kバイトが自由に使用できるエリアだということになります。このサイズは、プログラマが、プログラムテキストを作成したり、変数、配列を使うにつれて少なくなって行くことになります。

また、LIMIT 文によってBASICエリアとユーザエリアを分割した場合も、BASIC エリアを示すSIZE の値も減少することになります。

SIZE の内容は、PRINT SIZE 文によって調べることができます。

3.6.13 TI$

書 式 TI$

機 能 内蔵時計の示す時刻を表わす6桁のシステムストリング変数。 (TI：time)

解 説 MZ-2000は内蔵時計を持っており、秒単位で自動的に時を刻んでいます。TI$は、それが示す時刻を、時、分、秒それぞれ2桁の合計6桁のストリングで表わしています。たとえばTI$ の内容が"092035" であったとしたら、内蔵時計が午前9時20分35秒をその時に示していたことになります。

TI$は、MZ-2000の起動時に00時00分00秒でスタートします。

TI$を現在の実際の時刻に設定するには、ストリングの代入文を実行します。たとえば、午後7時の時報に合わせたい場合、あらかじめ、

TI$ = "190000"

とキー入力しておいて、時報と同時に、キャリッジリターンによってこの代入命令を実行させればよいことになります。

時間は00～23、分は00～59、秒も00～59の範囲で与えることはいうまでもありません。

3.7 音楽コントロール文

3.7.1 MUSIC

書式 MUSIC *x*$

x$…………ストリングデータ

機能 音楽の自動演奏を行います。

解説 音楽データは、音符を表現するストリングデータによってオペランドに記述します。1つの音符とは、どのオクターブの何という音で、どれだけの長さかを指定することによって決まります。即ち一般形は

1つの音符………〈オクターブの指定〉〈#（嬰音)〉音名〈音長〉

という並びとなります。かっこ〈〉内は省略されることがあります。

MUSIC文によって出力できる音は、は音（中央の1点ハの1オクターブ下の音）から3オクターブの半音階であり、音長は、全音符から32分音符までの各音長が指定できます。音楽データの実際の演奏では、更にTEMPO文によって決められる演奏速度が関係します。(音の強さをプログラム上で指定することはできませんが、音量は本体後部の音声ボリュームによって調整してください。)

■オクターブの指定

出力される音は3オクターブにわたっているのでまずどのオクターブ内の音かを指定します。即ち、図3－11に示すように中央のオクターブは無指定、下のオクターブは"－"符号で、上のオクターブは"＋"符号で区別します。

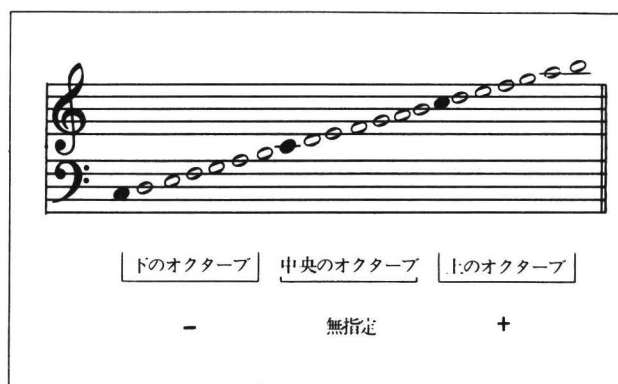



図 3－11

したがってたとえば図で黒で示した音はCの音（ハ調のド）ですが、3個のCの音は、オクターブの指定によって次のように区別されることになります。

下のオクターブのC …………－C

中央のオクターブのC…………C

上のオクターブのC …………＋C

■音の指定

音の指定は、音名と嬰音記号によって行います。音名は、CDEFGAB の各アルファベット大文字を、嬰音記号は、＃記号を用います。（シャープBASICでは、変音記号フラットは用いないのです。）休符は、Rによって示します。

各音の記号をピアノの鍵盤と対照して図3－12に示します。

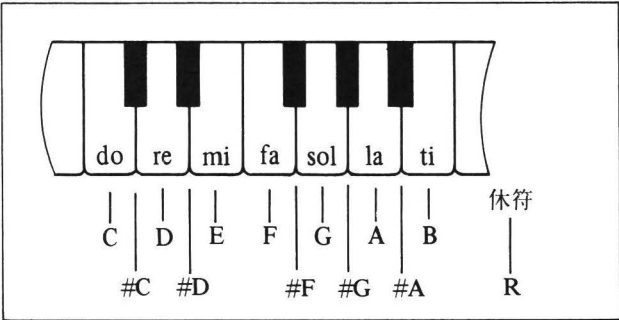



図　3－12

また"F"＝"＃E"，"C"＝"－＃B"，"＋C"＝"＃B"の関係があり、更に、3点ハの音（中央の1点ハの2オクターブ上の音）は"＋＃B"で出すことができます。

■音長の指定

オクターブと音名の指定された音に対して、次に音長を指定します。音長は音名につづいて、0～9の数字で指定します。数字と音符の関係を図3－13に示します。

図　3－13

同じ音長の音符が続くときは、2番目の音符からは音長の指定は省略できます。BASIC インタープリタの起動時には、音長のデフォルト値は5（♩の音）となっています。

例

図3-14に示す2オクターブのイ長調音階をA2＄に代入して鳴らしてみます。

```
10  A2＄＝"－A5－B3＃CDE＃F＃GA5B3＋＃C＋D＋E＋＃F＋＃G＋A8R5"
20  MUSIC  A2＄
```

図　3－14

3.7.2 TEMPO

書　式　TEMPO

x…………数値データ

機　能　MUSIC 文で演奏する音楽の演奏速度を決めます。

解　説　MUSIC 文の実行は、TEMPO 文によって指定されたテンポに従います。BASIC インタープリタの起動時には、テンポ＝4 （中位の速度）がデフォルト値となっていますが、*x* ＝1 ～ 7 の範囲で任意に設定し直すことができます。テンポ1が最も遅いテンポであり、テンポ7が最も早いテンポです。

TEMPO 1　………Lento, Adagio　（最も遅い）

∫

TEMPO 4　………Moderato　（中庸）

∫

TEMPO 7　………Molto Allegro, Presto　（最も早い）

例　図3-15に示す変ホ長調の旋律を演奏するプログラム例です。

```
10 VA$ = " + #D6+D3+F+ #DG+C"
20 VB$ = " #G7D5R3C"
30 VC$ = " #D6- #A3 #A5R3D"
40 VD$ = " F6C3+C5R3"
50 TEMPO 6
60 MUSIC VA$ , VB$ , VC$, VD$
```

図 3—15

†) J.Brahms (1833—1897) 作曲、ヴィオラソナタ、 Op.120-II 第1楽章より

3.8　グラフィックコントロール文

3.8.1　GRAPH

書　式　GRAPH　<I*a*, O*b*, C, F>

a……… グラフィックエリア番号：1、2または3

b……… グラフィックエリア番号：0、1、2、3、12、13、23または123

機　能　グラフィック表示について、入出力モードの設定、グラフィックエリアのクリアあるいはフィリング（埋めつくし）を行います。(GRAPH：graphic control)

解　説　MZ-2000本体内に、グラフィックボード（MZ-1R01……ページ1含む)、ページ2（MZ-1R02）およびページ3（MZ-1R02）をオプションで拡張することができます。

■グラフィックエリアの入力モードの設定

BASICのグラフィック表示文（GRAPH C、GRAPH F、SET、RESET、LINE、BLINE、PATTERNの各文）は、入力モードとなっているページへデータを送ります。

GRAPH I1………グラフィックメモリ／ページ1を入力モードとする
GRAPH I2………グラフィックメモリ／ページ2を入力モードとする
GRAPH I3………グラフィックメモリ／ページ3を入力モードとする

たとえば、次のマルチステートメント、

GRAPH I1：　SET 160、100

は、グラフィックメモリ／ページ1の中央のドットをセットします。グラフィックメモリ／ページ2およびページ3の内容は変わらず、また実際のCRTディスプレイ上の表示は、出力モードの状態によって決まるものであり、この文を実行したときに画面の中央にドットがセットされて見えるとは限らないわけです。(I：input)

■グラフィックエリアの出力モードの設定

グラフィックエリアとCRTディスプレイ上への実際の表示とのスイッチの働きをします。

GRAPH O1………グラフィックメモリ／ページ1を出力モードとする
GRAPH O2………グラフィックメモリ／ページ2を出力モードとする
GRAPH O3………グラフィックメモリ／ページ3を出力モードとする
GRAPH O12（またはO21)………ページ1、2を出力モードとする
GRAPH O13（またはO31)………ページ1、3を出力モードとする
GRAPH O23（またはO32)………ページ2、3を出力モードとする
GRAPH O123（順不同)……ページ1、2、3すべてを出力モードとする。
GRAPH O0………全ページとも出力しない

キャラクタメモリ（ビデオRAM）の内容は常にCRTディスプレイ上へ表示されているので、GRAPH O12を実行すると、3ページのビデオRAMの内容が同一画面上にオーバレイして表示されることになります。(O：output)

■グラフィックエリアのクリア

キャラクタ表示を［CLR HOME］＋［SHIFT］によってクリアするのと同様に、入力モードに設定されているグラフィックエリアの全ドットをリセットし、エリアをクリアします。

GRAPH C　　　　　(C：clear)

■グラフィックエリアのフィリング

クリアと逆に、入力モードに設定されているグラフィックエリアの全ドットをセットします。

GRAPH　F　　　　　　　(F：fill)

例　次のGRAPH文は、オペランドに6個のファンクションを置いています。各ファンクションはコンマ"，"で区切ります。

GRAPH　O0, I2, F, I1, C, O1

3.8.2　SET

書　式　SET　*x*, *y*

x………数値データ：x-座標

y………数値データ：Y-座標

機　能　入力モードに設定されているグラフィック エリア上の任意の1ドットをセットします。

解　説　セットするドット位置は、X-座標：*x*、Y-座標：*y*で指定します。グラフィックエリア上のX-Y座標は、図3－20で示すように、X方向は左から右へ0～319、(ハイリゾリューションモードの場合は0～639)　Y方向は上から下へ0～199の範囲となっています。座標位置の指定がグラフィックエリアを越えた場合は、次の数値内では無視され、エラーにはなりません。

0≦X-座標≦16383

0≦Y-座標≦16383

即ち、各種の連続した演算過程で、グラフィック表示エリアを越えたとしても、データオーバーによる中断は、この範囲で避けることができます。

例　図3-16はSET文を用いたグラフ表示の例です。

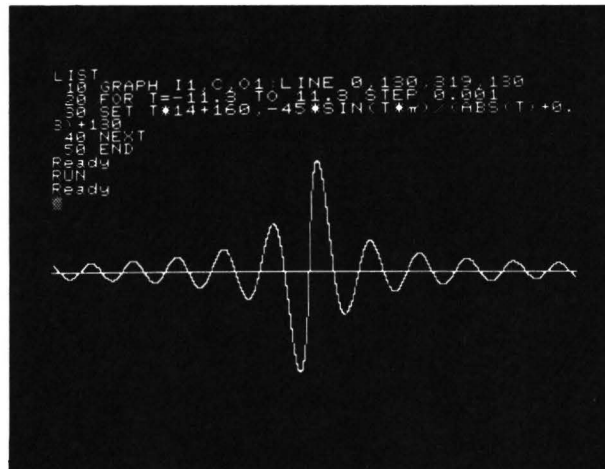



図　3－16

3.8.3 RESET

書　式　RESET *x*, *y*

x………数値データ：X-座標

y………数値データ：Y-座標

機　能　入力モードに設定されているグラフィックエリア上の任意の1ドットをリセットします。

解　説　リセットするドット位置と X-Y 座標の関係および、エリア外のデータで無視される値の範囲は、SET文の場合と同じです。

例　次のプログラムは、まず最初のGRAPH文でグラフィック表示を消した後、グラフィックエリア／ページ1を入力モードに設定して、エリア内の全ドットをセットします。そして、ページ1をグラフィック表示出力モードとしてから、座標点(200、160)を中心とする半径0、10、20、…………、150の同心円を描きます。図3－17にプログラムを実行したようすを示しており、円周がエリア外にはみ出してもエラーとならず無視されているのがわかります。

プログラム

```
10  GRAPH I1, F, O1
20  FOR R=0  TO 150  STEP 10
30  FOR TH=0  TO  2  STEP 0.01
40  RESET  R*COS(TH*π)+200, R*SIN(TH*π)+160
50  NEXT  TH, R
60  END
```

プログラムの実行

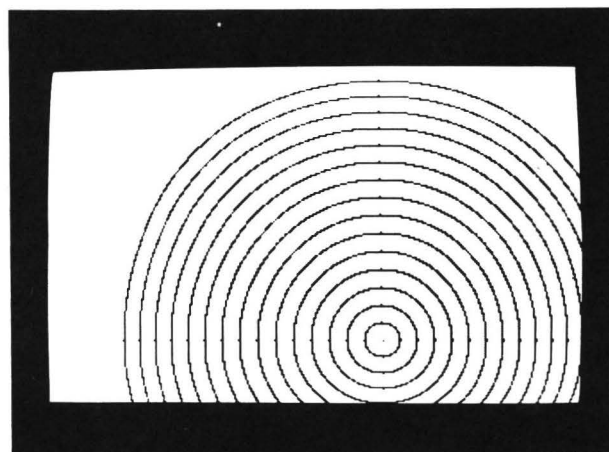



図　3－17

3.8.4 LINE

書　式　LINE　x_1, y_1, x_2, y_2 〈, x_3, y_3, ………, x_n, y_n〉

x_i………数値データ：X-座標

y_i………数値データ：Y-座標

機　能　入力モードに設定されているグラフィックエリア上に任意のラインを描きます。

解　説　LINE文のオペランドは、X-Y座標を示すx_i、y_iが組みとなって、最初の組みの示す座標点から次々に線分を引いて行きます。エリア内の座標位置と、エリア外の座標位置が前後に指定された場合も、表示エリア内で線分が描かれます。X-Y座標の指定範囲は、SET 文の場合と同じです。

例　図3－18の画面例は、１つの正方形を画面の中央に描いたもので、GRAPH文およびLINE文を直接実行命令として使っています。

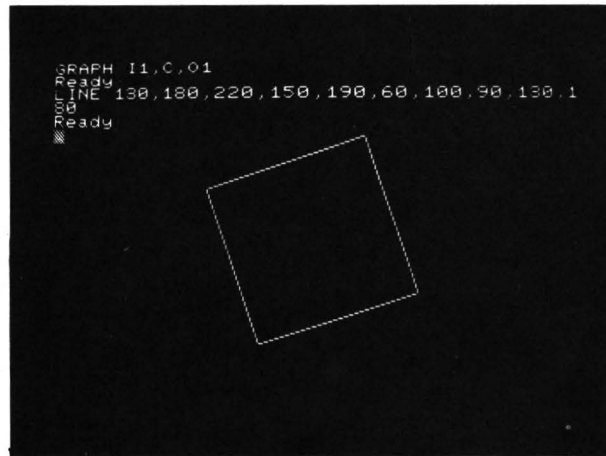



図　3－18

3.8.5 BLINE

書　式　BLINE x_1, y_1, x_2, y_2 〈, x_3, y_3, ………, x_n, y_n〉

x_i………数値データ：X-座標

y_i………数値データ：Y-座標

機　能　入力モードに設定されているグラフィックエリア上に任意のブラックライン(リセットされた線分)を描きます。(BLINE：black line)

解　説　描く線分がリセットされた線分であることの他は、LINE文の機能と同様です。

例　図3-19はBLINE文の実行例を示しています。画面の上方にプログラムリストを表示しておき、このプログラムをRUNさせます。すると行番号10によってグラフィックエリア／ページ1がフィルされ画面が真白になります。続いて、行番号20のBLINE文によって画面の上方からY-座標=90までクリアされ、元のプログラムリストが顔を出します。次に、行番号30および40によって下の部分に線模様を描きます。

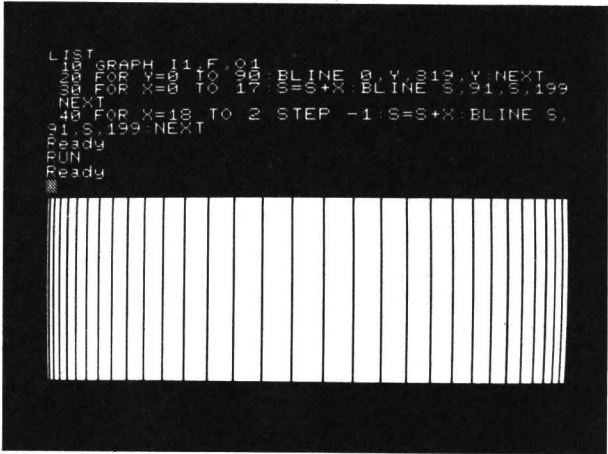



図　3—19

3.8.6 POSITION

書　式　POSITION　x，y

x………数値データ：X-座標

y………数値データ：Y-座標

機　能　PATTERN 文によるドットパターンの表示位置を決めるために、グラフィックエリア上のポジションポインタの位置を設定します。

解　説　PATTERN 文の実行は、ポジションポインタの示す座標位置から開始されます。それら各文の実行によってポジションポインタの示す座標位置も移動しますが、POSITION 文は、任意にその座標位置を設定する文です。

ポジションポインタは、X-Y座標について

$0 \leqq x \leqq 319$（ハイリゾリューションモードの場合は $0 \leqq x \leqq 639$）

$0 \leqq y \leqq 199$

の範囲で設定できます。図3-20参照。

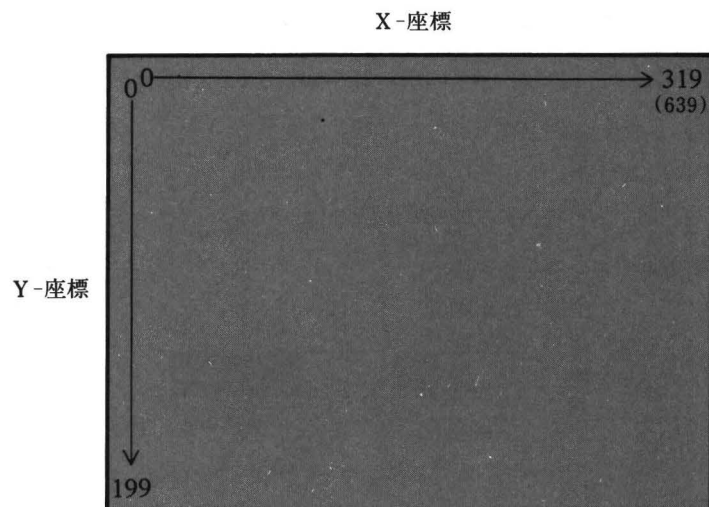



図　3－20

■ポジション・ポインタ

ポジションポインタとは、グラフィックエリア上の1ドット位置を示すポインタであり、PATTERN　文の実行の際に参照されるポインタです。この文の実行は、ポジションポインタの示すグラフィックエリア上の座標位置を起点として実行され、実行後は、その実行内容に従ってポジションポインタの位置も移ります。

グラフィックコントロールにおいて表示位置を管理する意味で、キャラクタ表示におけるカーソルの役割に似た働きをもっています。

3.8.7 PATTERN

書　式　PATTERN　x_1 , y_1 $ <, x_2 , y_2 $> ……… <, x_n , y_n $>

x_i ………数値データ：ドットパターンの積み重ね段数（±1～±24の範囲）

y_i $………ストリングデータ：8ビット単位のドットパターンを表わすデータ並び

機　能　入力モードに設定されているグラフィックエリア上に任意のグラフィックパターンを描きます。

解　説　グラフィックパターンは、入力モードのグラフィックエリア（GRAPH Ｉ文の実行によって決められる）上に、ポジションポインタの示す位置（POSITION 文の実行、あるいは、既に行っているPATTERN 文によって決められる）を起点として、積み重ね段数をx_i、ドットパターンのデータ並びをy_i $の内容として描かれます。PATTERN 文の実行によってポジションポインタも移動して行きます。

描くべきドットパターンは、８ビット単位で与え、その各ドットパターン要素は、ポジションポインタの示す座標位置からＸ軸の正の方向に、ドットのセット／リセットを決めます。８ビットをドットパターンの単位とすると、256通りの要素によってドットパターンを表現できることになります。ドットパターンを表わすデータ並びは、y_i $という形によって示されているようにストリングデータで表現します。

たとえば、８ドット全部をリセットする要素は、８桁の２進数$00000000_{(2)}=00_{(16)}=0_{(10)}$をストリングデータに変換したCHR $($00）または CHR$（0）で表現するし、８ドット全部をセットする要素は、$11111111_{(2)}=FF_{(16)}=255_{(10)}$をストリングデータにしたCHR $($FF）またはCHR $(255)、あるいは"π"（キャラクタ"π"のASCII コードは$255_{(10)}$です。付録Ａ.1のASCIIコード表を参照）でそれぞれ表現することができます。

図３－21に示したドットパターンの例では、黒くしたドットをセットするデータ要素は、ビットデータ$01001110_{(2)}=4E_{(16)}=78_{(10)}$をストリングデータ化した、CHR $($4E)、CHR $(78)または"Ｎ"によって表現されます。

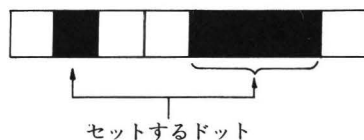



図　3－21

ドットパターン要素の積み重ねに関しては、先ず積み重ね方向は重ね段数x_iの符号によって決められ、下から上へ重ねるには正の値を、上から下へ重ねるには負の値とします。重み積ねは、上向き／下向きに$|x_i|$段実行したあとポジションポインタがＸ軸の正の方向へ８ドット分移り、先のドット表示のすぐ右隣りに続けて描かれます。ただし$1 \leq |x_i| \leq 24$でなければなりません。

たとえば、A$＝"ABCDEFG"（即ちCHR$(65)＋CHR$(66)＋………＋CHR$（71））の時、次の PATTERN 文を実行すると、図３－22に示すドットパターンが表示されることになります。

```
PATTERN　－5，A$
```

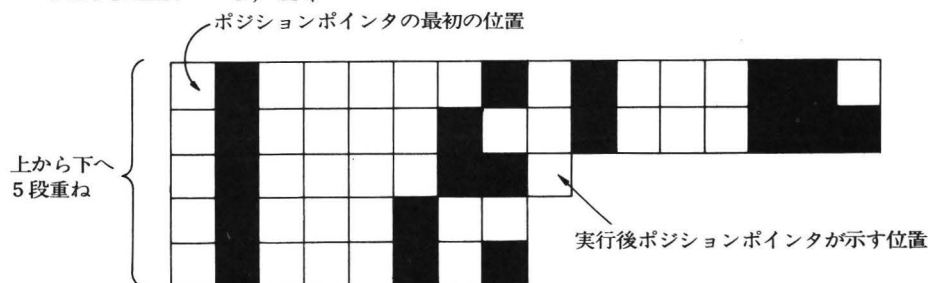



図　3－22

例

PATTERN文によって漢字や平仮名を表示させる例を取り上げてみます。漢字はアルファベットや数字などと較べてはるかに複雑な文字であり、ここでは16×16のドットマトリクスを漢字1文字の単位として表現することを考えてみます。

たとえば電算機の"電"という漢字を16×16ドットマトリクスで表現してみると図3−23のようになります。

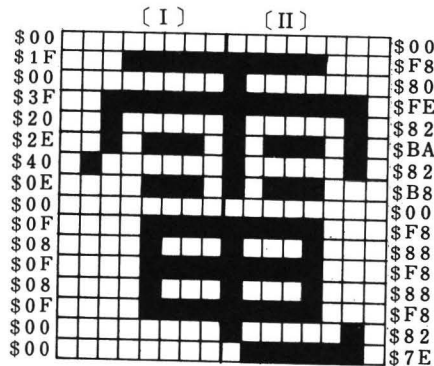



図 3−23 "電"を16×16ドットマトリックスで表現した例

図で中央に引いた太線は、積み重ねの並びを区切るもので、最初左側〔I〕の16段の積み重ねを描いておいて、続けて右側の〔II〕を描いて"電"を構成しようというものです。また最上段の1段と、両サイドのドット列は全てリセットと決めておき、隣り同士の文字がくっつかないように考えています。

図の両脇に16進コードを示していますが、これがデータとなります。PATTERN文によって、上から下へドットパターンを描くものとすると、"電"のストリングデータは、

CHR$($00)+CHR$($1F)+CHS$($00)+CHR$($3F)+CHR$($20)+
………+CHR$($88)+CHR$($F8)+CHR$($82)+CHR$($7E)

という32個のキャラクタの並びとなります。

このデータがたとえば、DE$(20)というストリング配列要素に代入されていたとすると、

PATTERN −16, DE$(20)

によってポジションポインタ位置から"電"が表示されることになります。

図3−24の左側は、漢字データ作成プログラムによってカーソルエディションを用いたデータ作成の一例を示しています。右側はそうして作成した、漢字データの1グループと、平仮名をデータファイルから呼び出してCRTディスプレイ上に表示させたものです。

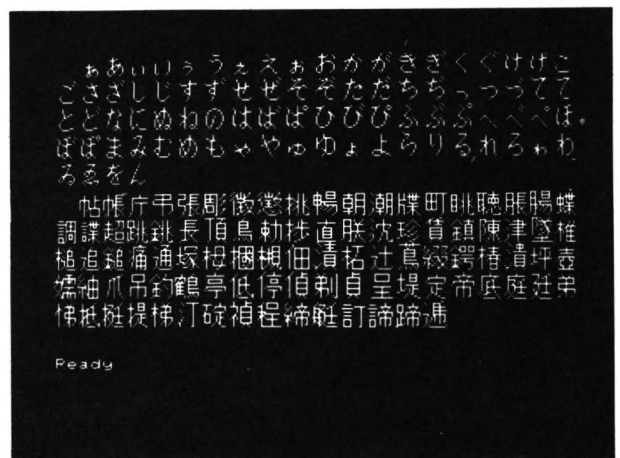
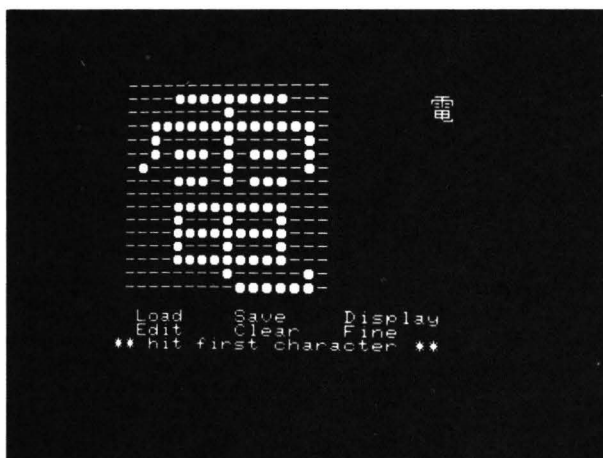



図 3−24

3.8.8 POINT

書　式　POINT　(x, y)

x………数値データ：X-座標

y………数値データ：Y-座標

機　能　x、yで指定されるグラフィックエリア上の任意の座標点がセットされているか、リセットされているかの情報を読み出す関数です。この関数の値はページ1、2および3を装備している場合次のように0、1、2、3、4、5、6、7のいずれかの整数となります。

POINT(x, y)	POINT情報
0	グラフィックエリア／ページ1、2、3ともリセットされている
1	グラフィックエリア／ページ1のみセットされている
2	グラフィックエリア／ページ2のみセットされている
3	グラフィックエリア／ページ1、2のみセットされている
4	グラフィックエリア／ページ3のみセットされている
5	グラフィックエリア／ページ1、3のみセットされている
6	グラフィックエリア／ページ2、3のみセットされている
7	グラフィックエリア／ページ1、2、3すべてがセットされている

ただし、グラフィックボード(MZ-1R01……ページ1のみ含む)のみ装備している場合はこの関数の値は次のようになります。

POINT(X，Y)	POINT情報
0、2、4または6	ページ1がリセットされている
1、3、5または7	ページ1がセットされている

また、グラフィックボード(MZ-1R01)にグラフィックエリア／ページ2(MZ-1R02)を追加装備している場合は、この関数の値は、次のようになります。

POINT(X，Y)	POINT情報
0または4	ページ1、2がリセットされている
1または5	ページ1のみセットされている
2または6	ページ2のみセットされている
3または7	ページ1、2ともセットされている

この関数によって、グラフィック表示データをフィードバックすることができ、例えば、曲線と曲線の交点をグラフィック表示によって調べるといった応用が考えられます。

例　次の文は、POINT関数の値によって多岐分岐する例です。

```
100　ON　POINT　(X，Y)　GOTO　210，220，230
```

3.8.9 POSH

書式　POSH

機能　グラフィックエリア上のポジションポインタの、現在のX-座標（水平位置）を与えるシステム変数。
（POSH：horizontal position）
従って、POSHのとり得る値は、

0≦POSH≦319

です。
ただし、ハイリゾリューションモード（640×200ドット／画面）の場合、POSHのとり得る値は、

0≦POSH≦639

となります。

3.8.10 POSV

書式　POSV

機能　グラフィックエリア上のポジションポインタの、現在のY-座標（垂直位置）を与えるシステム変数。
（POSV：vertical position）
従って、POSVのとり得る値は、

0≦POSV≦199

です。

3.9 データファイル入出力文

BASICプログラムテキストは、SAVEコマンドによってカセットテープ上に記録、保存ができ、またLOADコマンドによって再びBASICテキストエリア上へ読み出すことができますが、各種のデータをカセットテープ上に記録、保存、再呼び出しを行うことができるよう以下に説明するデータファイル入出力文が用意されています。

3.9.1 WOPEN／T

書　式　WOPEN<／T> <*file name*>

機　能　カセットテープ上にシーケンシャルデータファイルを作成するために、カセットテープファイルをオープンします。(WOPEN：write open)

解　説　WOPEN文は、シーケンシャルデータファイルの書き出しオープン（write open）を宣言する文であり、 *file name* は作成するシーケンシャルデータファイルのファイル名になります。
WOPEN文を実行すると、PRINT／T文によって、数値データまたはストリングデータを順次カセットテープ上に記録して行きますがCLOSE文が実行されてはじめて、カセットテープ上に、そのシーケンシャルデータファイルをファイルエンドマークと共に正式に登録します。*file name* を記述しないと、ファイル名のないデータファイルが作成されてしまいます。データファイルの内容を明示するようなファイル名を必ず書いておくようにすると、ファイルがたくさん作成されたとき間違いが少なくなります。

3.9.2 PRINT／T

書　式　PRINT／T d_1 <, d_2, d_3, ………, d_n>

d_i………数値データまたはストリングデータ

機　能　WOPEN文によってオープンされているカセットデータファイルに、オペランドに記述した各種データを順次書き込んで行きます。

解　説　オペランドに記述する出力データは、数値データ、ストリングデータいずれも自由に置くことができます。1つのPRINT／T文に複数個のデータを置く場合は、セパレータ"，"を用います。
実際のデータ書き込みは、256バイト単位で行われます。PRINT／T文を実行中にプログラムのブレークを行ったり、CLOSE文を実行せずに置いた場合、不完全なカセットデータファイルのままカセットテープデッキ内に残ってしまうことになります。ROPEN文のもとでINPUT／T文を実行させた場合、あるデータ以降が正しく呼び出せないということが生じてしまいます。従って、オープンしたファイルは正しくクローズしなくてはなりません。
また、ファイルオープン中にカセットテープを交換したり、巻き戻し、送りなどの操作を行うと正常なファイルコントロールが不可能になるので行ってはなりません。

例

次のプログラムは、カセットデータファイル"Name list"を作成する例です。１クラス50人の生徒名が、出席番号Ａの順に１次元配列Ｎ＄（Ａ）の中に既に代入されているものとします。

```
300  WOPEN／T  "Name list"
310  FOR  A＝1  TO  50：PRINT／T  N＄（A）：NEXT
320  CLOSE／T
```

3.9.3 CLOSE／T

書　式

CLOSE〈／T〉

機　能

WOPEN 文を実行している場合、WOPEN をクローズして、カセットテープ上に１つのシーケンシャルデータファイルを作成します。

ROPEN文を実行している場合は、ROPEN をクローズして、新たに別のファイルのオープンができるようにします。

3.9.4 ROPEN／T

書　式　ROPEN <／T>　<*file name*>

機　能　カセットテープ上のシーケンシャルデータファイルに登録されているデータを読み出すためカセットファイルをオープンします。（ROPEN：read open）

解　説　ROPEN文は、シーケンシャルデータファイルの読み出しオープンを宣言する文であり、*file name*は読み出しオープンするシーケンシャルデータファイルのファイル名を指定します。

ROPEN文を実行すると、INPUT／T文によってカセットテープ上のデータを順次変数または配列要素に代入することができます。

データの読み出しを終了したらCLOSE文を実行してファイルを閉じます。

*file name*を記述しないと最初に見つかったBASICシーケンシャルデータファイルを読み出しオープンします。

3.9.5 INPUT／T

書　式　INPUT／T　v_1 <, v_2, v_3, ………, v_n>

v_i………数値変数、数値配列要素またはストリング変数、ストリング配列要素

機　能　ROPEN文によってオープンされているカセットシーケンシャルデータファイルから順次データを読み出します。

解　説　INPUT／T文を実行すると、そのオペランドに記述されている入力並びの変数または配列要素に順次カセットファイル上のデータを読み出し代入します。従って、カセットファイル上のデータ並びとINPUT／T文の入力並びのデータ型はそれぞれ一致していなくてはなりません。

シーケンシャルデータファイル上の数値データがINPUT／T文でストリング変数（または配列要素）に対応された場合は、ストリングデータとして代入されます。この場合数値データが、5.17という値だとしたら"　5.17"というストリングとして扱われます。

INPUT／T文の実行中にデータ切れ、即ちアウトオブファイルが発生したら＊Error 63 (Out of file)となります。

ファイル・エンドの扱い

作成されたシーケンシャルデータファイルからデータを読み出す場合アウト・オブ・ファイルが発生するとエラー(Error 63)が発生してBASICコマンドレベルへ戻ってしまいます。ファイルデータのデータ数がわかっている場合は、読み出し回数をそれに合わせれば良いのですが、データ個数が不明で、アウト・オブ・ファイルになってもプログラムを停止しないようにするには、シーケンシャルデータファイルを作成する時に最後に必ず終端を示すデータを入れておくようにすれば、それによってエンド・オブ・ファイルを判断することができます。たとえば、文字通り"END OF FILE"というストリングデータを書いておく方法が考えられます。

3.10 機械語プログラムコントロール文

ここで解説するステートメントグループは、特殊な処理を行うために機械語（machine language）プログラムを作成してBASICプログラムとのリンクを行ったり、モニタプログラムエリア等を直接アクセスして必要な処理を行ったりするためのステートメントグループです。

3.10.1 LIMIT

書　式　LIMIT　*ad*

ad………アドレス：数値データ、もしくは4桁の16進数

機　能　BASICで使用するメモリエリアを、*ad*で指定するアドレスまでに制限し、その次のアドレスから$FFFF（65535）番地までをユーザエリアとします。

あるいは、LIMIT　MAX文によって全エリアをBASICエリアに戻します。

解　説　BASICで、機械語プログラムとのリンクを行う場合、あるいは特殊なデータをメモリ上に構成しようとする場合は、それに必要なユーザエリアをこのLIMIT文で確保して、BASICで使用するエリアと区別しておかなければなりません。

2つのエリアを分割するのに、BASICで使用するエリアの最後のアドレスをオペランドに記述しますが、そのアドレス値は、10進数または16進数で直接指定します。

BASICの使用エリアをメモリの最大に戻すには、LIMIT　MAX文を実行します。

付録A・3のメモリマップを参照して下さい。

例　LIMIT　$DFFF………BASICプログラムエリアを$DFFF番地までに制限し、それ以降をユーザエリアとします。図3－25参照。

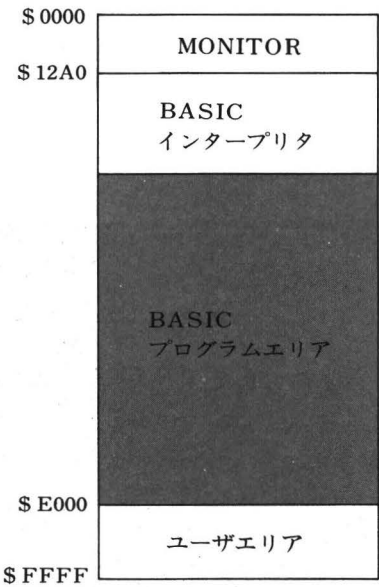



図　3－25

3.10.2 POKE

書　式　POKE *ad*, *d*

　ad ………アドレス：数値データ、もしくは4桁の16進数

　d ………数値データ：0～255、もしくは2桁の16進数

機　能　メモリ上の任意のアドレスにデータをストアします。

解　説　*ad*で指定するメモリ位置に直接1バイトのデータを書き込む文でありアクセスするアドレスは任意です。

データは1バイトデータであるので、0～255（16進の＄00～＄FF）のいずれかであり、数値データであれば0～255の範囲のいずれかの整数でなければなりません。

POKE文は、LIMIT文と関係なくメモリ空間上の任意の位置に対して実行可能ですからBASICやMONITOR本体を壊す可能性があるので、その使用には充分注意が必要です。

LIMIT文によって確保したユーザ・エリアは、BASICによって使用されることはないので、機械語プログラムまたはデータをPOKE文で構成する場合、あらかじめLIMIT文を実行しておくようにします。

3.10.3 PEEK

書　式　PEEK (*ad*)

　ad………アドレス：数値データ、もしくは4桁の16進数

機　能　数値データ*ad*の値の示すメモリ上のアドレスの内容を与えます。メモリ上のデータは8 bit データであるので、0から255の範囲の結果が与えられます。

*ad*の値は、0から65535までの範囲でなければなりません。メモリ上のアドレスを直接指定する場合は、16進数表現が可能です。

逆に、メモリ上の指定アドレスにデータをストアするにはPOKE文を用います。

例　図3-26は、PEEK関数の使用例を示しています。アドレス4544から4559、即ち TAB キーデータのエリアの内容を表示させたものです。これによって現在 TAB キーは、8、16、24、32、40、48、56文字という風に8文字ごとにタビュレーションが設定されていることがわかります。

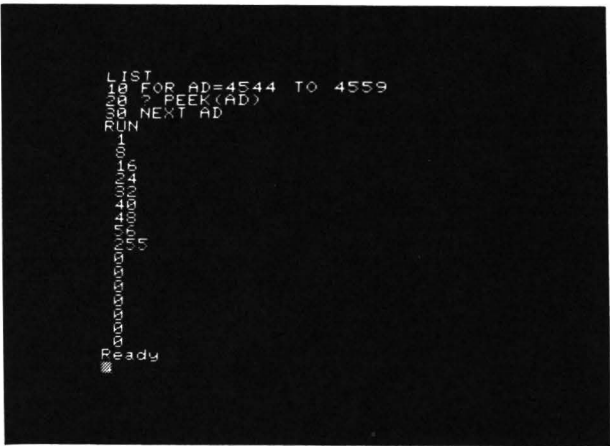



図 3−26

3.10.4 USR

書　式　USR (*ad*)

ad………アドレス：数値データ、もしくは4桁の16進数

機　能　数値データ*ad*の値の示すメモリ上のアドレスにプログラムのコントロールを移します。これは機械語のサブルーチンへの分岐命令、CALL *ad* と同じです。従って機械語プログラム中でリターン命令（RET、RET　cc）があると、USR関数を実行した次のステートメントへコントロールが戻ります。

*ad*の値は、0から65535までの範囲でなければなりません。メモリ上のアドレスを直接指定する場合は、16進表現が可能です。

書　式　USR（*ad*, *x*＄）

ad………アドレス：数値データ、もしくは4桁の16進数

x＄………ストリングデータ

機　能　USR関数で、アドレスデータとともに、ストリングデータを与えた場合、機械語プログラムのCALLを実行する直前に、ストリングデータ*x*＄の内容がストアされているメモリエリアの先頭アドレスを、CPUのDEレジスタにセットし、*x*＄の長さをBCレジスタにセットします。

この関数によって、BASICプログラム中で使用しているストリングデータを、機械語プログラムに受け渡すことができます。

3.11 プリンタコントロール文

プリンタコントロールの詳細については、各プリンタのマニュアルによってください。

3.11.1 PRINT／P

書　式　PRINT／P　〈e_1 d_1 e_1 d_2 ……e_n d_n〉

　e_i…………出力並び：　数値データ、ストリングデータ、コントロールコード

　d_i…………セパレータ

機　能　出力並びで指定したデータの値をプリンタへ出力します。

解　説　PRINT／P文は、PRINT文の実行とほぼ同じ形式でデータ表示をプリンタ上へ行います。セパレータ"，"と"；"の働き、TAB関数の働きも同様です。

ただし、プリンタには各種の印字モードがあり、プリントされる文字の形や行間隔などが切り換えられるようになっていること、1文字を表現するドットマトリクスの構成などの違いがあります。

次に、プリンタコントロールで用いる特殊コードを示します。

PRINT／P　CHR＄(5)　…………フォーム・フィード（改頁）を行う

PRINT／P　CHR＄(6)　…………動作モードの初期化を行う

PRINT／P　CHR＄(16)　…………行間にスペースを空けるプリントモードにする

PRINT／P　CHR＄(17)　…………行間圧縮のプリントモードにする

PRINT／P　CHR＄(18)　…………倍文字プリントモードにする

PRINT／P　CHR＄(19)　…………倍文字プリントモードを解除する

PRINT／P　CHR＄(20)　…………縮少文字プリントモードとする

PRINT／P　CHR＄(21)　…………縮少文字プリントモードを解除する

3.11.2 IMAGE／P

書　式　IMAGE／P　x＄

　x＄…………ストリングデータ：8ビット単位のドットパターンを表わすデータ並び

機　能　出力並びで指定した任意のドットパターンをプリンタへ出力します。

解　説　グラフィックコントロールのPATTERN文に似た働きのある、グラフィックイメージをプリントする文です。プリンタのヘッドは上下方向に並んでおり、グラフィックイメージのプリントは、PATTERN文によるCRTディスプレイ上のグラフィックパターン表示を90°傾けたものになります。

たとえば、次の文の実行によって図3－27に示すイメージプリントが行われます。

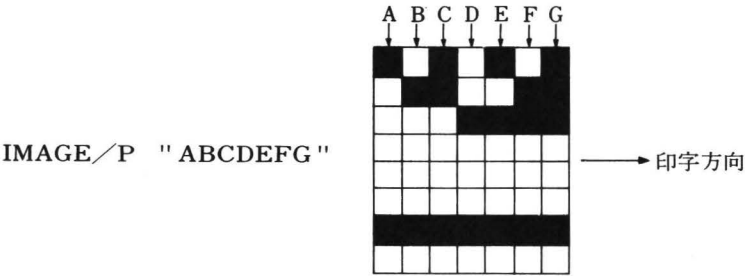



図　3－27

3.11.3　COPY／P

書　式　COPY／P *n*

n………ディスプレイ表示ページ指定：１、２、３、４、５、６、７または8

機　能　CRT ディスプレイ上のデータ表示をそのまま１画面ぶんプリンタ上へコピーします。

解　説　CRT ディスプレイ上のデータ表示はキャラクタ表示、グラフィックエリア１、2、3のグラフィック表示とがあり、COPY／P 文では、オペランドの記述によってどの表示データをコピーするか指定します。

キャラクタ表示のコピーをとる場合

キャラクタ表示のコピーをとる場合は次の COPY／P 文を実行します。

COPY／P　1

グラフィック表示のコピーをとる場合は、グラフィックエリア上の各ドットの SET／RESET の状態を調べて、イメージモードと同様の方法でドットパターンのデータを出力しプリントします。

グラフィックエリアは1と2と3があるので、次の7通りのCOPY／P文が記述できます。

COPY／P　2………グラフィックエリア１の表示データをプリンタ上にコピーします。

COPY／P　3………グラフィックエリア２の表示データをプリンタ上にコピーします。

COPY／P　4………グラフィックエリア１、2を同時に表示した場合の表示データをプリンタ上にコピーします。

COPY／P　5………グラフィックエリア3の表示データをプリンタ上にコピーします。

COPY／P　6………グラフィックエリア１、３を同時に表示した場合の表示データをプリンタ上にコピーします。

COPY／P　7………グラフィックエリア２、３を同時に表示した場合の表示データをプリンタ上にコピーします。

COPY／P　8………グラフィックエリア１、２、３を同時に表示した場合の表示データをプリンタ上にコピーします。

COPY／P文は、GRAPH　O文とは関係なく使用できます。

CONSOLE　GH命令により640×200ドット／画面のハイリゾリューションモードに設定している時に80桁ラインプリンタにてグラフィック表示のコピーをとる場合は、かならずプリンタの印字モードを縮少文字プリントモードに設定してください。136桁ラインプリンタを使用する場合は、普通印字モードでコピー可能です。

(例)　100　PRINT／P　CHR＄(20)

　　　110　COPY／P　8

3.11.4　PAGE／P

書　式　PAGE／P *x*

x………数値データ

機　能　プリンタの印字で、何行を１ページとするかを決める文です。

解　説　プリンタの初期状態では１ページ66行印字されますが、それを １～255　の範囲で任意に決めることができます。　たとえば、

PAGE／P　22

文によって１ページを22行と決めておくと、プリンタの"TOP　OF　FORM"ボタンを押すか、ページ送りのコードを送ると (PRINT／P　CHR＄(５)文による) 22行単位で次の先頭行へプリント用紙が送られます。

3.12　I/Oポートアクセス文

3.12.1　INP

書　式　INP@　*p*, *v*

p………ポート番号：０～255 もしくは＄00～＄FF

v………数値変数または数値配列要素

機　能　I／Oポート上にある１バイトのデータを変数（または配列要素）に入力します。(INP：input)

解　説　INP 文の機能は、Z80A-CPU の"IN"命令の機能に似ており、オペランド@ *p* の *p* をアドレス・バスの下位(A_0からA_7）に割り付け、指定可能256 ポートの１つを選択してそのポート上にある１バイトのデータが取り込まれます。CPU の"IN　A、(n)"命令では、アキュムレータに１バイトのデータが取り込まれますが、BASIC の INP 文では、変数 *v* に取り込まれます。データは１バイトなので、数値変数または数値配列要素 *v* に0～255(または $00～$FF)のいずれかの数値が代入されます。

ユニバーサルI／Oカードを介して外部機器からのデータ入力を行うには、カード上のポート設定スイッチによって定めたポート番号によってアクセスを行います。

3.12.2　OUT

書　式　OUT@　*p*, *x*

p………ポート番号：０～255 もしくは＄00～＄FF

x………数値データ：0～255もしくは$00～$FF

機　能　I／Oポート上に１バイトのデータを出力します。

解　説　OUT文の機能は、Z80A-CPUの"OUT"命令の機能に似ており、出力ポートを選択して、そこへ１バイトデータを出力します。CPUの"OUT　(n)、A"命令では、アキュムレータの内容が該当ポートへ出力されますが、BASICのOUT文では、数値データの形で１バイトデータが出力されます。従って、数値データは、0～255（または＄00～＄FF）のいずれかの数値でなければなりません。

MZ-2000本体は、16進で＄E0 ポート以降を各種コントロールで使用しており、これらのポートについては、Owner's Manual を参照の上注意して扱わなくてはなりません。

BASIC MZ-1Z001の関数

Chapter
4

この章は、BASICインタープリタMZ-1Z001の関数について、数値関数、ストリング処理関数、タビュレーションコントロール関数の各組み込み関数（built-in function）順にそれぞれの機能を解説しています。

グラフィック処理、機械語プログラム処理等で特殊に用いられる各関数については、前章の各ステートメントグループの中で解説されていますから、そちらを参照してください。

どの組み込み関数も、あらかじめ定義する必要なく任意に呼び出すことができます。

4.1 組み込み数値関数

4.1.1 ABS

書 式 ABS(x)

x………数値データ：数値定数、数値変数、数値配列要素またはそれらの算術式

機 能 数値データxについて、その絶対値｜x｜を与えます。(ABS：absolute value)即ち、

$x \geqq 0$ のとき ABS(x)＝x

$x < 0$ のとき ABS(x)＝$-x$

となります。

例 PRINT ABS（3－8）

```
 5
Ready
```

図4－1は、ABS関数の機能をグラフィック表示で示した例です。

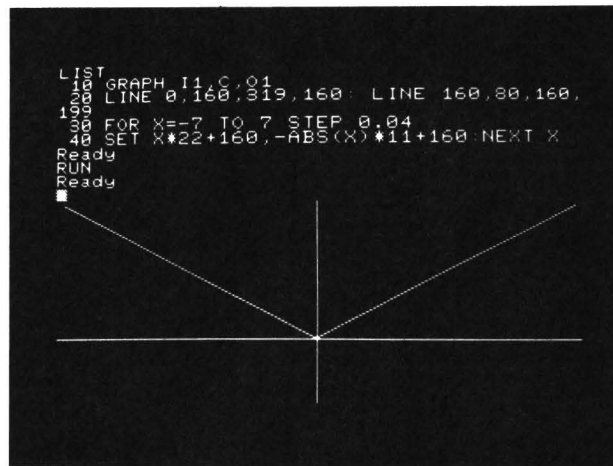



図 4－1

4.1.2 INT

書　式　INT(x)

機　能　数値データxについて、xを越えない最大の整数を与えます。(INT : integer)

例　図4－2は、INT関数による整数化によって、連続的に変化する量が階段状の離散値に変換されるもようを示しています。

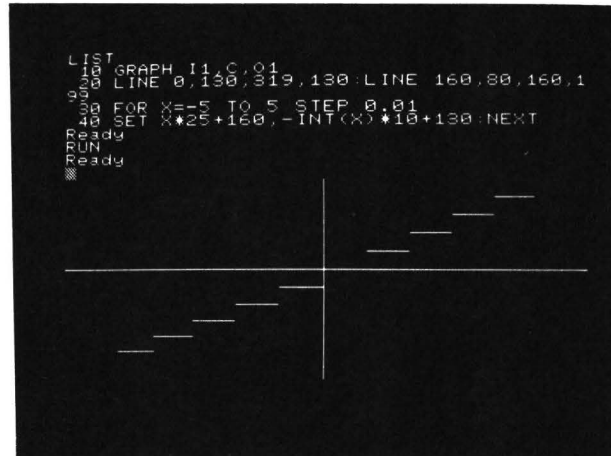



図　4－2

4.1.3 SGN

書　式　SGN(x)

機　能　数値データxについて、符号関数sign xの値を与えます。(SGN : sign)即ち、

$x>0$　のとき　SGN(x)＝1

$x=0$　のとき　SGN(x)＝0

$x<0$　のとき　SGN(x)＝－1

となります。

例　図4－3は、SGN関数の働きを、図4－2のINT関数の働きに対照させて示しています。

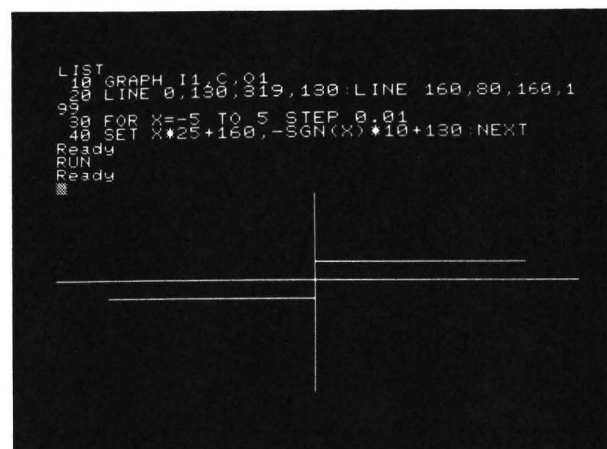



図　4－3

4.1.4 SQR

書　式　SQR(x)

機　能　数値データxについて、その平方根 $\sqrt{x}$ を与えます。(SQR: square root)

xの値は正または0でなければデータエラーとなります。

例　プログラム

```
10  X=5
20  PRINT  X , SQR(X)
30  X=X-1 :  GOTO  20
```

プログラム実行

```
RUN
 5              2.236068
 4              2
 3              1.7320508
 2              1.4142136
 1              1
 0              0
-1
*Error  3  in 20…………Xの値が負になったためデータエラーが発生した
Ready
```

4.1.5 SIN

書　式　SIN(x)

機　能　数値データxについて、三角関数(trigonometric function) $\sin x$の値を与えます。ただし、xの単位はラジアン(radian)です。(SIN : sine)

度(degree)単位の数値データdについて、三角関数 $\sin d^\circ$ の値を得るには、

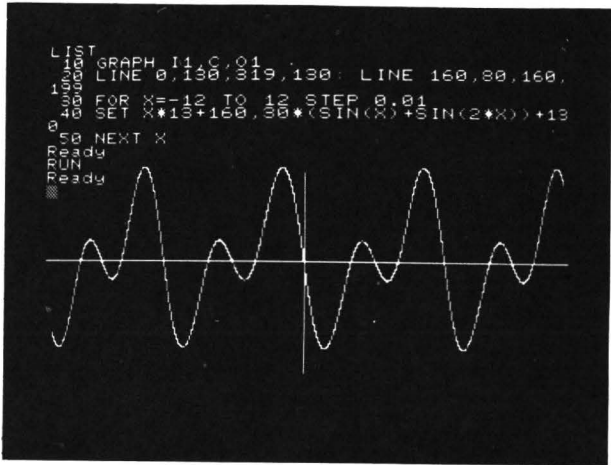
　　SIN(d∗π／180)

とします。

例　図4－4は、－12ラジアンから12ラジアンまで角度xを変化させたときの、$y=\sin x+\sin 2x$ のグラフを示しています。

図　4－4

4.1.6 COS

書　式　COS (x)

機　能　数値データxについて、三角関数cos x の値を与えます。ただし、xの単位はラジアン(radian)です。(COS：cosine)

度(degree)単位の数値データdについて、三角関数 cos d°の値を得るには、

COS (d＊π／180)

とします。

例　図4－5は、COS関数を組み合わせて1つの繰り返し曲線を描かせたものです。

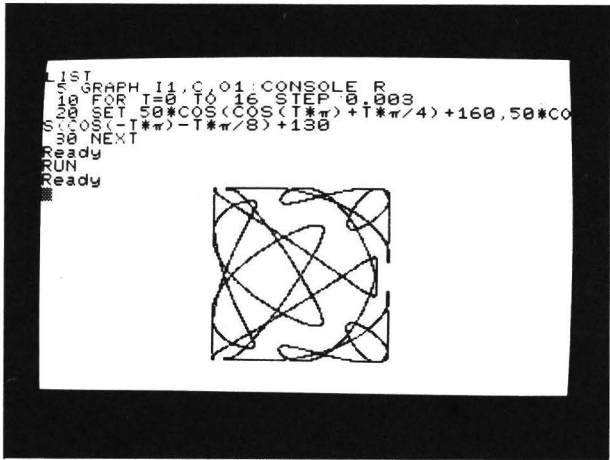



図 4―5

この画面表示は、CONSOLE R 文によって画面全体の白黒が反転されています。従ってSET文によって描いた曲線は、黒く描かれています。

4.1.7 TAN

書式 TAN(*x*)

機能 数値データ*x*について、三角関数 tan *x* の値を与えます。ただし、*x*の単位はラジアン(radian)です。(TAN : tangent)

度(degree)単位の数値データ*d*について、三角関数tan d° の値を得るには、

TAN($d*\pi/180$)

とします。

tan *x*は、$x\to\frac{\pi}{2}\pm n\pi$、(n＝0、1、2……) のとき｜tan *x*｜→＋∞となり、TAN関数も*x*の値によって結果がオーバフローすることがあります。

例 図4－6は、TAN関数のオーバフローを調べるために、89.99999°から0.000001°刻みで90°に近付けた様子を示しています。89.999999°でオーバフローが発生したことがわかります。

図 4－6

4.1.8 ATN

書式 ATN(*x*)

機能 数値データ*x*について、逆三角関数(inverse trigonometrical function) arctan *x*の値を与えます。ただし得られる結果の単位はラジアン(radian)です。(ATN : arctangent)

arctan *x*は、無限多価関数(infinite many-valued function) ですが、ATN関数では、その主値(principal value)を与え、結果は$-\pi/2$から$\pi/2$の間の値となります。

度(degree)単位の結果を得るには、

ATN(x)$*180/\pi$

とします。

例 PRINT ATN(1)*180／π ;"°"

45°

Ready

4.1.9 EXP

書　式　EXP(x)

機　能　数値データxについて、指数関数(exponential function)e^xの値を与えます。(EXP：exponential function)

eは自然対数の底(the natural logarithmic base)です。

例　PRINT　EXP(1)，EXP(2)

```
  2.7182818                    7.3890561
Ready
```

図4－7は正規分布形のグラフを表示させた例です。

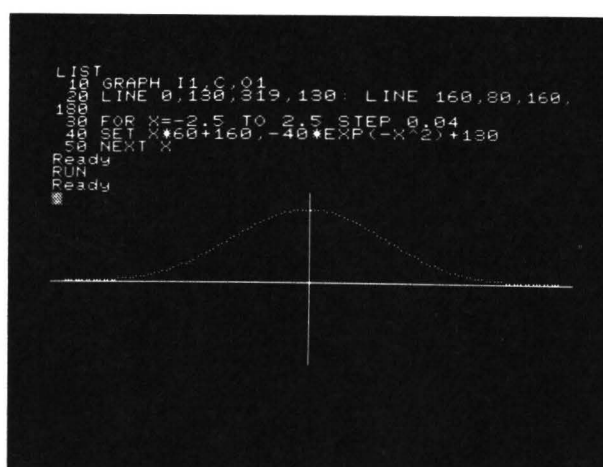



図　4－7

4.1.10 LOG

書　式　LOG(x)

機　能　数値データxについて、常用対数(common logarithm) $\log_{10} x$の値を与えます。真数xは、0より大きくなければなりません。(LOG：logarithm)

B＞0かつB≠1であるとき、数値データAの値について、Bを底とする対数関数の値(the value of the logarithm of A with the base B) $\log_B A$は、

LOG(A)／LOG(B)

または、

LN(A)／LN(B)

によって得られます。

4.1.11 LN

書　式　LN(x)

機　能　数値データxについて、自然対数(natural logarithm) $\log_e x$ の値を与えます。真数xは、0より大きくなければなりません。(LN：natural logarithm)

数学公式にあるように、真数の積は、対数の和で表わすことができます。即ち、

LN(x＊y)＝LN(X)＋LN(Y)

LN(x／y)＝LN(X)－LN(Y)

LOG(x＊y)＝LOG(X)＋LOG(Y)

LOG(x／y)＝LOG(X)－LOG(Y)

一般に、演算実行時間は、左辺の方が短くなります。

4.1.12 RND

書　式　RND(x)

機　能　数値データxについて、0.00000001から0.99999999までの値をとる擬似乱数(pseudo random numbers) を与えます。(RND：random numder)

数値データxの値によって次の2通りの働きを持っています。

x＞0とした場合は、BASICインタープリタが発生する擬似乱数系列で、前回与えた乱数にひきつづく、次の乱数を与えます。

x≦0とした場合は、RND関数による擬似乱数系列の発生を初期化し、先頭の擬似乱数値を与えます。この初期化操作によって、乱数によるシミュレーションなどに再現性を持たせることができます。

4.2　ストリング処理関数

4.2.1　LEFT$

書　式　LEFT$ (*x*$, *n*)

x$…………ストリング定数、ストリング変数、ストリング配列要素またはそれらの結合式

n …………数値定数、数値変数、配列要素またはそれらの算術式

機　能　ストリングデータ*x*$について、左から*n*文字で構成するストリングを与えます。

*n*は、０から255の範囲でなければなりません。

もし*n*≧LEN(*x*$) ならばLEFT$ (*x*$, *n*) =*x*$となります。またn＝０のときは、LEFT$ (*x*$, *n*) ＝" " (null string) となります。

例

```
PRINT  LEFT$(" Personal Computer MZ-2000" ，17)
Personal Computer
Ready
```

プログラム

```
  10  A$＝"ナツメ  ソウセキ"
  20  B$＝LEFT$(A$ , 3)
  30  PRINT  "セイハ  " ; B$ ; " デス"
```

プログラム実行

```
RUN
セイハ  ナツメ   デス
Ready
```

4.2.2　MID$

書　式　MID $ (*x* $, *m* , *n*)

機　能　ストリングデータ*x*$について、左から*m*番目の文字から*n*文字で構成するストリングを与えます。(MID : middle)

*m*は、１から225の範囲、*n*は０から255の範囲でなければなりません。

もし*n*>LEN(*x*$) −mならば、MID$ (*x*$, *m* , *n*) =RIGHT$ (*x* $ ，LEN (*x* $) − m +1)

即ち*x*$のm文字から終わりまで全部となります。また*n*＝０であればMID$ (*x*$, *m* , *n*) ＝" " (null string) となります。

例　プログラム

```
  10  A$＝"Wolfgang Amadeus Mozart"
  20  INPUT N
  30  PRINT  MID$(A$ , N , 8)
```

プログラム実行

```
RUN
?  10
Amadeus
Ready
```

4.2.3 RIGHT$

書　式　RIGHT$ ($x$$, n)

機　能　ストリングデータ$x$$について、右からn文字で構成するストリングを与えます。

nは、0から255の範囲でなければなりません。

もし$n \geqq$ LEN ならば RIGHT$ ($x$$, n) = $x$$ となります。また$n = 0$のときは、RIGHT$ ($x$$, n) = " " (null string) となります。

例　プログラム

```
10  A$="ナミメ  ソウセキ"
20  B$=RIGHT$(A$ , 4)
30  PRINT  "ナマエハ  "; B$ ; "  デス"
```

プログラム実行

```
RUN
ナマエハ  ソウセキ  デス
Ready
```

4.2.4 SPACE$

書　式　SPACE$ (n)

機　能　数値データnについて、n個の連続したスペースストリングを与えます。

このストリング処理関数は主に、PRINT文で用いられ、項目の削除（文字をスペースで消す）やタビュレーション操作に使用されます。

例　プログラム

```
10  PRINT  "┌" ; STRING$ ("─" , 30) ; "┐"
20  PRINT  "|" ; SPACE$ (11) ; "Table  1  " ; SPACE$ (11) ; "|"
30  PRINT  "└" ; STRING$ ("─" , 30) ; "┘"
```

プログラム実行

```
RUN
┌──────────────────────────────┐
│           Table  1           │
└──────────────────────────────┘
Ready
```

4.2.5 STRINGS$

書　式　STRING$ (*x*$, *n*)

機　能　ストリングデータ*x*$について、その先頭の文字を連続して*n*個つないで構成するストリングを与えます。

例　前記のSPACE$関数は、このSTRING$関数の*x*$が" "であった場合の関数と考えることができます。

プログラム

```
10 R$=STRING$("*",10)
20 PRINT R$;" Table 1 ";R$
```

プログラム実行

```
RUN
**********  Table 1  **********
Ready
```

4.2.6 CHR$

書　式　CHR$ (*x*)

機　能　数値データ*x*について、ASCIIコードが*x*である1つのキャラクタを与えます。(CHR : character) 従って*x*は、0から255 (16進数の$00から$FF) の範囲でなければなりません。CHR$ (*x*) 関数とは逆に、キャラクタのASCIIコードを得るには、ASC関数が用いられます。

例　図4—8は、ASCII コード30〜255に相当する、すべてのキャラクタをテレビ画面上に表示させたものです。

ASCIIコード0〜29は特殊コードとなっています。付録のASCIIコード表 (P.98〜99) を参照してください。

図 4—8

4.2.7 ASC

書　式　ASC(*x*$)

機　能　ストリングデータ*x*$について、先頭のキャラクタに対応するASCIIコードを与えます。(ASC：A-SCII)キャラクタとASCIIコードの関係は、巻末のASCIIコード表を参照してください。

ASC関数とは逆に、ASCIIコードからキャラクタを得るには、CHR$関数が用いられます。

例　プログラム

```
10 INPUT "オオモジ ヲ ニューリョク セヨ：";CL$
20 FOR N=1 TO LEN(CL$)
30 PRINT CHR$(ASC(MID$(CL$,N,1))+32);
40 NEXT N
50 END
```

プログラム実行

```
RUN
オオモジ ヲ ニューリョク セヨ：FLOCCINAUCINIHILIPILIFICATION†)
floccinaucinihilipilification
Ready
```

4.2.8 STR$

書　式　STR$(*x*)

機　能　数値データ*x*について、その値を表わすストリングを与えます。

数値データが指数表現となる場合には、指数表記のストリングを与えます。

STR$関数とは逆に、数値を表わすストリングをそのまま数値データに変換するには、VAL関数が用いられます。

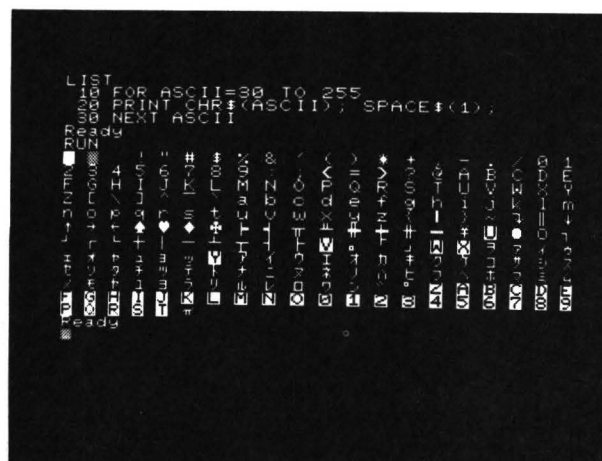
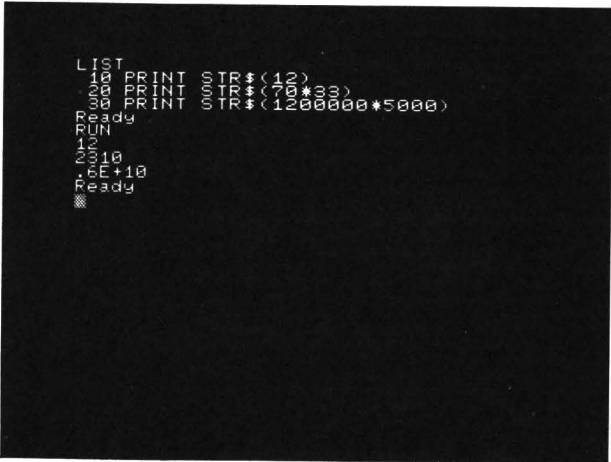
例　図4－9はSTR$関数を用いたデータの変換の例を示しています。正の数値の表示では、正号"＋"が略された1つのブランクが数値の前に表示されますが、STR$関数でストリングに変換したものには、このブランク(space)は付加されないことにも注意してください。

図 4－9

†) floccinaucinihilipilification：軽視

4.2.9 VAL

書　式　VAL (x＄)

機　能　ストリングデータ$x$$について、それが表わす数値データを与えます。(VAL : value)

VAL関数とは逆に、数値データをストリングにそのまま変換するには、STR$関数が用いられます。

例　図4－10は、VAL関数を用いたデータの変換の例を示しています。日常よく行うように、1000単位でコンマ"，"を打って数値を入力する方法を許し、それをVAL関数によって実際の数値データに変えるように工夫した例です。

```
LIST
 10 V$="68,800,777,000"
 20 V=VAL(LEFT$(V$,2)+MID$(V$,4,3)+MID$(
V$,8,3)+RIGHT$(V$,3))
 30 PRINT V
Ready
RUN
 .68800777E+11
Ready
```

図　4—10

4.2.10 LEN

書　式　LEN($x$$)

機　能　ストリングデータ$x$$について、その構成キャラクタ数を与えます。(LEN : length)

キャラクタのうち、CRTディスプレイに表示されないスペース"　"やコントロール用ストリングデータもそれぞれ1文字として数えられます。

例

```
PRINT  LEN(STR$(π))
  9
Ready
```

システム変数πは、8桁の定数3.1415927と小数点を含んでおり、STR$関数でストリングデータとしたときストリングの長さは9となったのです。

```
PRINT  LEN("Personal Computer MZ-2000")
 25
Ready
```

4.2.11 CHARACTER$

書　式　CHARACTER$ (x , y)

x…………数値データ：X―座標

y…………数値データ：Y―座標

機　能　現在のキャラクタ表示で、X―座標がx、Y―座標がyで指定される画面上の位置に表示されている1つのキャラクタを与えます。

キャラクタ表示は、ASCIIコードが30～255 のものが表示可能ですから、CHARACTER$ 関数によって得られるキャラクタは、CHR$ (30) ～CHR$(255) のいずれかとなります。

座標指定の範囲は、80キャラクタ表示モード、40キャラクタ表示モードでそれぞれ次のように限られます。

■80キャラクタモードの場合

X―座標：$0 \leqq x \leqq 79$

Y―座標：$0 \leqq y \leqq 24$

■40キャラクタモードの場合

X―座標：$0 \leqq x \leqq 39$

Y―座標：$0 \leqq y \leqq 24$

例　図4 －11は、CRTディスプレイ上の上部5行に表示されているメッセージを CHARACTER$ 関数によって読み出して、文字間にスペースを空けて、ディスプレイの下部へ表示し直したものです。

```
SHARP

Personal Computer

MZ-2000

LIST
 10 FOR Y=0 TO 5:FOR X=0 TO 19
 20 CURSOR X*2,Y+15
 30 PRINT CHARACTER$(X,Y)
 40 NEXT X,Y
 50 END
Ready
RUN

S H A R P

P e r s o n a l   C o m p u t e r

M Z - 2 0 0 0

Ready
```

図　4－11

4.3 タビュレーションコントロール関数

4.3.1 TAB

書　式　TAB(*x*)

x…………数値データ：数値定数、数値変数、数値配列要素またはそれらの算術式

機　能　PRINT文のオペランドに使用され、データ表示のポインタであるカーソルを、数値データ*x*で指定するタビュレーション位置へ移動します。（TAB：tabulation）

CRT ディスプレイの左端が、タビュレーション＝0、右端が79（80キャラクタモードの場合）あるいは39（40キャラクタモードの場合）です。

タビュレーションによるカーソルの移動は、X座標の正の方向にだけ移動でき、現在のカーソル位置より左側へのタビュレーション値が指定された場合は無効（何も実行しない）となります。

付　　録 Appendix

この付録には、次のものが置かれています。

■ASCIIコード表………表A・1

■BASICインタープリタ MZ-1Z001エラーメッセージ表………表A・2

この表には、プログラム、コマンド実行中に起こり得る全てのエラーを載せています。BASICインタープリタは、エラーの発生をこの表に示されたエラー番号を表示することによって知らせます。

■メモリ・マップ

■三角関数、双曲線関数表

A.1 ASCIIコード表

MZ-2000システムのASCIIコード表を次に示します。

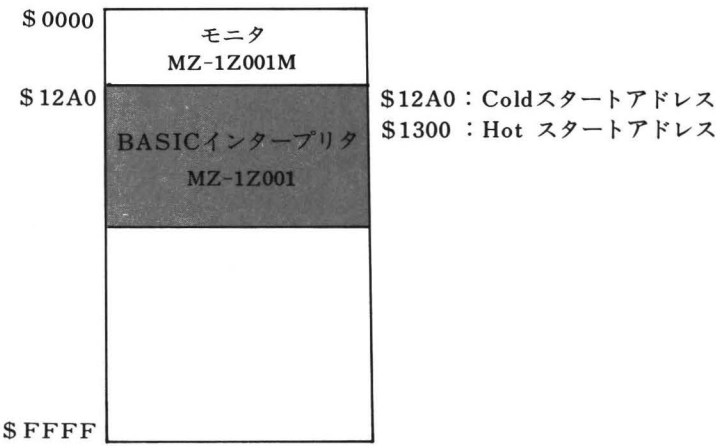
10進	16進	キャラクタ	10進	16進	キャラクタ	10進	16進	キャラクタ	10進	16進	キャラクタ	10進	16進	キャラクタ
0	00	NULL	26	1A		52	34	4	78	4E	N	104	68	h
1	01	↓	27	1B		53	35	5	79	4F	O	105	69	i
2	02	↑	28	1C		54	36	6	80	50	P	106	6A	j
3	03	→	29	1D		55	37	7	81	51	Q	107	6B	k
4	04	←	30	1E	■	56	38	8	82	52	R	108	6C	l
5	05	HOME	31	1F	▒	57	39	9	83	53	S	109	6D	m
6	06	CLR	32	20		58	3A	:	84	54	T	110	6E	n
7	07	DEL	33	21	!	59	3B	;	85	55	U	111	6F	o
8	08	INST	34	22	"	60	3C	<	86	56	V	112	70	p
9	09	GRPH	35	23	#	61	3D	=	87	57	W	113	71	q
10	0A	SFT LOCK	36	24	$	62	3E	>	88	58	X	114	72	r
11	0B		37	25	%	63	3F	?	89	59	Y	115	73	s
12	0C	カナ	38	26	&	64	40	@	90	5A	Z	116	74	t
13	0D		39	27	'	65	41	A	91	5B	[	117	75	u
14	0E	L SCRIPT	40	28	(	66	42	B	92	5C	＼	118	76	v
15	0F	カナ CANCEL	41	29	)	67	43	C	93	5D	]	119	77	w
16	10		42	2A	*	68	44	D	94	5E	^	120	78	x
17	11		43	2B	+	69	45	E	95	5F	￣	121	79	y
18	12		44	2C	,	70	46	F	96	60	`	122	7A	z
19	13		45	2D	-	71	47	G	97	61	a	123	7B	{
20	14		46	2E	.	72	48	H	98	62	b	124	7C	|
21	15		47	2F	/	73	49	I	99	63	c	125	7D	}
22	16		48	30	Ø	74	4A	J	100	64	d	126	7E	~
23	17		49	31	1	75	4B	K	101	65	e	127	7F	↴
24	18		50	32	2	76	4C	L	102	66	f			
25	19		51	33	3	77	4D	M	103	67	g			

10進	16進	キャラクタ	10進	16進	キャラクタ	10進	16進	キャラクタ	10進	16進	キャラクタ	10進	16進	キャラクタ
128	80	┃┃┃	154	9 A	│	180	B4	エ	206	CE	ホ	232	E8	H
129	81	↓	155	9 B	─	181	B5	オ	207	CF	マ	233	E9	I
130	82	↑	156	9 C	┴	182	B6	カ	208	D0	ミ	234	EA	J
131	83	→	157	9 D	┬	183	B7	キ	209	D1	ム	235	EB	K
132	84	←	158	9 E	┤	184	B8	ク	210	D2	メ	236	EC	L
133	85	♠	159	9 F	├	185	B9	ケ	211	D3	モ	237	ED	M
134	86	♥	160	A0	V	186	BA	コ	212	D4	ヤ	238	EE	N
135	87	◆	161	A1	。	187	BB	サ	213	D5	ユ	239	EF	O
136	88	♣	162	A2	「	188	BC	シ	214	D6	ヨ	240	F0	Ø
137	89	╞	163	A3	」	189	BD	ス	215	D7	ラ	241	F1	1
138	8A	╡	164	A4	W	190	BE	セ	216	D8	リ	242	F2	2
139	8B	╥	165	A5	X	191	BF	ソ	217	D9	ル	243	F3	3
140	8C	╨	166	A6	ヲ	192	C0	タ	218	DA	レ	244	F4	4
141	8D	╬	167	A7	ァ	193	C1	チ	219	DB	ロ	245	F5	5
142	8E	╪	168	A8	ィ	194	C2	ツ	220	DC	ワ	246	F6	6
143	8F	╫	169	A9	ゥ	195	C3	テ	221	DD	ン	247	F7	7
144	90	≡	170	AA	ェ	196	C4	ト	222	DE	゛	248	F8	8
145	91	¥	171	AB	ォ	197	C5	ナ	223	DF	゜	249	F9	9
146	92	U	172	AC	ャ	198	C6	ニ	224	E0	Z	250	FA	P
147	93	●	173	AD	ュ	199	C7	ヌ	225	E1	A	251	FB	Q
148	94	○	174	AE	ョ	200	C8	ネ	226	E2	B	252	FC	R
149	95	┐	175	AF	ッ	201	C9	ノ	227	E3	C	253	FD	S
150	96	┘	176	B0	Y	202	CA	ハ	228	E4	D	254	FE	T
151	97	┌	177	B1	ア	203	CB	ヒ	229	E5	E	255	FF	π
152	98	└	178	B2	イ	204	CC	フ	230	E6	F			
153	99	┼	179	B3	ウ	205	CD	ヘ	231	E7	G			

A.2 エラーメッセージ表

エラー番号		エラーの内容
1	Syntax	文法上の誤り
2	Overflow	数値データが許容範囲外、演算結果がオーバフローした
3	Illegal data	規定外の数値、変数が使われた
4	Data type	データと変数の型が一致しない
5	String length	ストリングの内容が255文字を越えた
6	Memory capacity	メモリ容量不足となった
7	Array definition	同じ配列変数を前より大きく定義しようとした
8	Line length	1行の長さが制限を越えた
9		
10	GOSUB nesting	GOSUB文のネスティングが16を越えた
11	FOR～NEXT nesting	FOR～NEXT文のネスティングが16を越えた
12	DEF FN nesting	DEF FN文による関数定義のネスティングが6を越えた
13	NEXT	対応するFOR文のないNEXT文が使われた
14	RETURN	対応するGOSUB文のないRETURN文が使われた
15	Undefined function	定義されていない関数が使われた
16	Undefined line number	存在しない行番号を参照しようとした
17	Can't continue	CONT文によるプログラムの継続ができない
18	Memory protection	BASICインタープリタの管理エリア内への書き込み要求をした
19	Instruction mode	ダイレクトコマンドとステートメントを混同して使った
20		
21		
22		
23		
24	READ	対応するDATA文のないREAD文が使われた
25		
26		
27		
28		
29		
30		
31		
32		
33		
34		
35		

エラー番号		エラーの内容
36		
37		
38		
39		
40		
41		
42		
43	Already open	すでにオープンされているファイルを更にオープンしようとした
44		
45		
46		
47		
48		
49		
50		
51		
52		
53		
54		
55		
56		
57		
58		
59		
60		
61		
62		
63	Out of file	ファイルデータ読み込みで、アウトオブファイルが起きた
64		
65	Printer is not ready	プリンタが接続されていないか、OFF 状態である
66	Printer hardware	プリンタにメカトラブルが起きた
67	Out of paper	プリンタ用紙ぎれ
68		
69		
70	Check sum	チェックサムエラー

A.3 メモリ・マップ

A.4 三角関数と双曲線関数

組み込み数値関数にない三角関数と双曲線関数およびそれらの逆関数を定義するための例を次に示します。使用する場合には各関数の定義域に注意する必要があります。

secant	FNA (X)＝1/COS(X)
cosecant	FNB (X)＝1/SIN(X)
cotangent	FNC (X)＝1/TAN(X)
arcsine	FND (X)＝ATN(X/SQR(1－X^2))
arccosine	FNE (X)＝－ATN(X/SQR(1－X^2))＋π/2
arcsecant	FNF (X)＝ATN(SQR(X^2－1))＋(SGN(X)－1)＊π/2
arccosecant	FNG (X)＝ATN(1/SQR(X^2－1))＋(SGN(X)－1)＊π/2
arccotangent	FNH (X)＝－ATN(X)＋π/2
hyperbolic sine	FNI (X)＝(EXP(X)－EXP(－X))/2
hyperbolic cosine	FNJ (X)＝(EXP(X)＋EXP(－X))/2
hyperbolic tangent	FNK (X)＝－EXP(－X)/(EXP(X)＋EXP(－X))＊2＋1
hyperbolic secant	FNL (X)＝2/(EXP(X)＋EXP(－X))
hyperbolic cosecant	FNM(X)＝2/(EXP(X)－EXP(－X))
hyperbolic cotangent	FNN (X)＝EXP(－X)/(EXP(X)－EXP(－X))＊2＋1
arc-hyperbolic sine	FNO (X)＝LN(X＋SQR(X^2＋1))
arc-hyperbolic cosine	FNP (X)＝LN(X＋SQR(X^2－1))
arc-hyperbolic tangent	FNQ (X)＝LN((1＋X)/(1－X))/2
arc-hyperbolic secant	FNR (X)＝LN((SQR(1－X^2)＋1)/X)
arc-hyperbolic cosecant	FNS (X)＝LN((SGN(X)＊SQR(X^2＋1)＋1)/X)
arc-hyperbolic cotangent	FNT (X)＝LN((X＋1)/(X－1))/2

MONITOR編

MONITOR編目次

はじめに……105

第1章　MONITOR MZ-1Z001Mコマンドとサブルーチンの使い方……107

1.1　MONITOR　MZ-1Z001Mの機能……108
1.2　モニタコマンドの機能と使い方……110
　1.2.1　Mコマンド……110
　1.2.2　Dコマンド……114
　1.2.3　Jコマンド……116
　1.2.4　Sコマンド……117
　1.2.5　Vコマンド……120
　1.2.6　Lコマンド……121
1.3　モニタ・サブルーチン……123

付　録……127

A.1　Z80A-CPU インストラクションセット対照コード表……128
　(ニーモニックコードのアルファベット順)
A.2　Z80A-CPU インストラクションセット対照コード表……138
　(オペコードの16進順)
A.3　IPLプログラム・アセンブリ・リスト……148
A.4　MONITOR MZ-1Z001Mのプログラムリスト……164
A.5　BASICテープのコピー作成と改訂の方法……205

はじめに

このマニュアルは、パーソナルコンピュータMZ-2000の標準システムソフトウェアMONITOR MZ-1Z001Mの命令仕様とサブルーチンの機能を解説しており、機械語（machine language）によってプログラムを作成したり、データを構成する場合、どのようにしたらよいかが示されています。

MONITOR MZ-1Z001Mは、MZ-2000のシステムソフトウェアの標準モニタプログラムであり、BASICインタープリタ、DISK BASICインタープリタ、倍精度DISK BASICインタープリタなどをバックアップする働きがおもな役割となっていますが、モニタへコントロールを移すことによって、独立した機械語モニタ（machine language monitor）として使用することができます。これによって、BASICプログラムの機械語サブルーチンを直接作成したり、簡単なデバッグを行うことができるだけでなく、自由にシステムプログラムを作成したり、カセットファイルを作成したり、独自の方法でMZ-2000をコントロールすることができます。

巻末には参考のためMONITOR MZ-1Z001Mの全アセンブリ・リストを掲載しております。

MONITOR MZ-1Z001M
コマンドとサブルーチンの使い方

Chapter 1

この章は、6つのモニタコマンドの機能と使い方および主要なモニタサブルーチンの機能と使い方とを解説しています。モニタプログラムによってプログラマは、機械語プログラムの作成、実行、簡単なデバック操作、さらに機械語プログラムのカセットテープ上へのファイル化が行えます。作成した機械語プログラムをBASICプログラムテキストのUSR関数によってユーザサブルーチンとして呼び出しリンクさせることもできます。

1.1 MONITOR MZ-1Z001Mの機能

MONITORプログラムはパーソナルコンピュータ MZ-2000の最も基本的なシステムプログラムの1つであり、その働きは幾通りかに分けて考えることができます。

まず第1に、その名前の通りMZ-2000の各種のシステムソフトウェア、たとえばBASICインタープリタやPASCALインタープリタ、FDOS等のいわゆるモニタ機能があり、基本的な論理操作を行う各種サブルーチンや、MZ-2000のハードウェア装置であるキーボード、CRTディスプレイ、カセットテープデッキ、タイマ回路、サウンド発生回路等のI/Oをコントロールするサブルーチンがモニタ中にまとめられており、BASICインタープリタなどのメインとなるシステムソフトウェアは、随時それらのサブルーチンをコールして処理を進めて行きます。従ってたとえばBASICのSAVEコマンドやLOADコマンド、PRINT文、MUSIC文、INPUT文等の実行の際には、該当する入出力デバイスをコントロールするモニタサブルーチンが図1－1に示すようにコールされるのです。

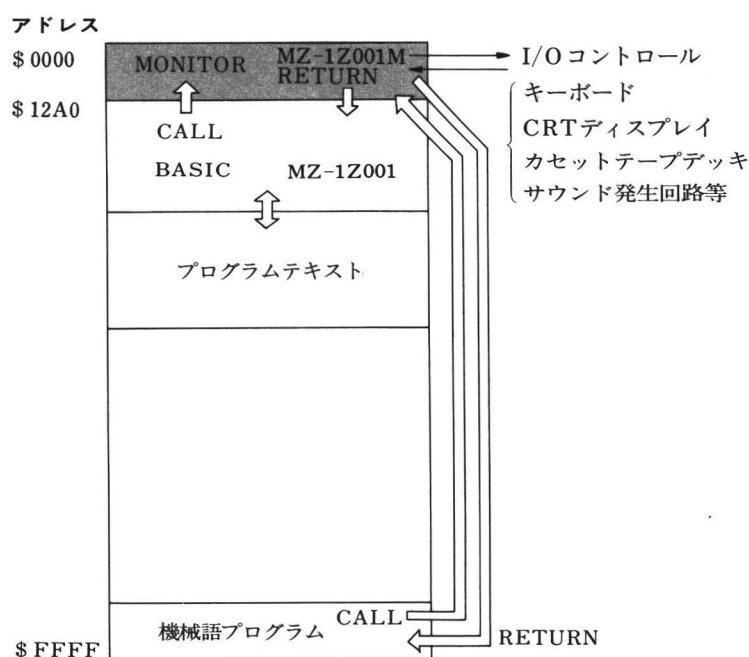



図 1－1 モニタサブルーチンのコール

この図には、メインメモリの後部にある機械語プログラムがモニタサブルーチンをコールしている矢印も描かれていますが、このように、ユーザプログラム中で直接使用することも勿論できます。

各種コントロールルーチンをBASIC等のメインプログラム中に含めることをせず1つの独立した、しかも汎用性のあるモニタ・プログラムとして構成しているのはMZ-2000処理システムをより系統的にするという目的によっているからです。系統化によってモニタプログラムには上記のようにハードウェアに直接結びついた各種のコントロールが中心にまとめられ、ソフトウェアとハードウェアの中間部という意味の、いわゆるファームウェアとしての働きが主体となっているのです。

MONITOR MZ-1Z001Mの大きさは約4.5Kバイトあり、図1－1に示されているようにメインメモリの先頭$0000番地からストアされます。

第2の働きはモニタ・コマンド・レベルへコントロールを移してやることによって簡便な機械語プログラムモニタの役割を果たすことです。即ち、単にメインシステムプログラムやユーザプログラムで共通に使うサブルーチンの集合というのではなく、メインシステムプログラムからモニタへシステムコントロール自体を移すことができ、次にあげる6つのモニタコマンドによって機械語によるプログラムの作成、データの作成、ファイル入出力といったアクティブなシステムコントロールができるのです。

Mコマンド……メモリの内容を書き替える。(memory correction)
Dコマンド……メモリブロックの内容を表示させる。(memory dump)
Jコマンド……任意のアドレスにコントロールを移す。プログラムの実行。(jump)
Sコマンド……メモリブロックの内容をカセットテープファイルにセーブする。(save)
Vコマンド……メモリブロックの内容とカセットテープファイルの内容を比較する。(verify)
Lコマンド……カセットテープファイルをロードする。(load)

図1－2はBASICコマンドMONの実行例であり、実行後アステリスク"＊"の表示に続くカーソル点滅によってコントロールがモニタ・コマンド・レベルへ移ったことが示されています。

図 1−2 BASICコマンドMONの実行

モニタ・プログラムはRAM上で働くので、モニタ・コマンドによって自分自身を自由に変えることができます。

たとえば、CPUのリスタート命令の応答や、割り込み応答を使う場合コールされる＄0000～＄0038番地、＄0066番地の内容等を任意に設定したり、モニタ・サブルーチンの機能を変えてしまうことができます。

また、カセットテープへの書き込み／読み出しが自由なので、モニタ・プログラム自身を含んだ1つの独自の機械語プログラムをファイリングすることができます。このような操作を行うさいは巻末付録A.4のモニタ・プログラム・アセンブリ・リストが参考になるでしょう。

1.2　モニタコマンドの機能と使い方

この節では、6個のモニタコマンドについて、それぞれの機能と使い方の解説を行います。各コマンド操作の一般的な取り扱い規約は以下のようになっています。

■コマンドあるいはデータのキー入力は、所定の欄に正しく行い、 CR キーを押すことによって実行されます。スペース等の不必要なデータを混じえてキー入力するとモニタコマンドはそのデータを受け付けません。

■データ入力あるいはデータ表示はすべて16進数表示で行われます。従って1バイトデータは16進2桁、1つのアドレスデータは16進4桁で扱われます。たとえば表示データが"15"であったとすれば16進数$15（10進数の21）を表わしているし、キー入力を行う場合、たとえば10進数10をデータとして入力する場合は所定の位置に"0A"ときちんと桁数を合わせてキー入力しなければなりません。（上位桁の"0"を省略することはできません）

■ファイル名の指定などで、モニタが受けつける以上の文字数を入力した場合、その数を越す文字列は無効となります。

■各コマンドの使用を終え、モニタコマンド待ちに戻すには BREAK を押します。

■各コマンドのアクセス範囲にはプロテクションが一切ありません。従って、モニタプログラム自身のリストアップ、変更等の操作が自由に行えることになりますが、逆に、システムコントロールを行うプログラムを壊してしまいCPUの暴走を引き起こすことも充分あり得ることになり、そのような操作を行う場合は細心な配慮を払う必要があります。

1.2.1　Mコマンド(M：memory correction)

機　能　指定するアドレスから順々に1バイトずつメモリ・コレクション（データ書き換え）を行います。

解　説　Mコマンドを与えると図1－3に示す表示がなされます。

図　1−3

ここでモニタは、メモリ・コレクションを開始するアドレス指定を要求しています。たとえば、＄70A0 番地から、メモリ・コレクションを行うには、＄記号につづけて"70A0"とキー入力して CR キーを押します。すると図1－4に示す表示がなされます。

図 1−4

ここでモニタは、＄70A0 番地に現在セットされているデータ 00 を表示し、このデータを書き換えるかどうかの指定を要求しています。データを書き換えるには、16進2桁――即ち、00～FFのいずれか――でカーソル位置から続けてキー入力します。
たとえば、データ＄C9 (RET 命令のオペコード) をセットするには、図1－5のようにカーソル位置から続けて2桁にこのコードを入力して、CR キーを押します。すると＄70A0番地のメモリ・コレクションを終えて、次に＄70A1番地のメモリ・コレクションに移ります。

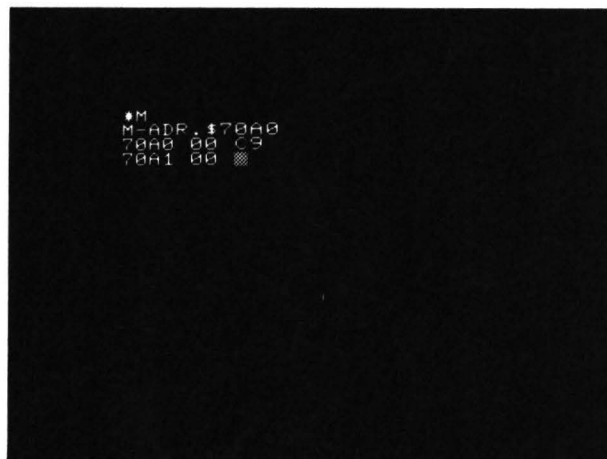



図 1−5

もし、その番地上のデータを書き換える必要がなければ CR キーを押すことによって、データをそのままにして、次の番地のメモリ・コレクションへ移ります。たとえば $70A1 番地の内容を00のままにするため、CR キーを押すと図1－6のように次の $70A2 番地へ移ります。

図 1—6

カーソル位置からの2桁に、16進数以外の文字が与えられると、コマンドを中断して、新たにアドレス指定を要求して来ます。図1－7は、たとえば" S "というキーを押した場合を示しています。

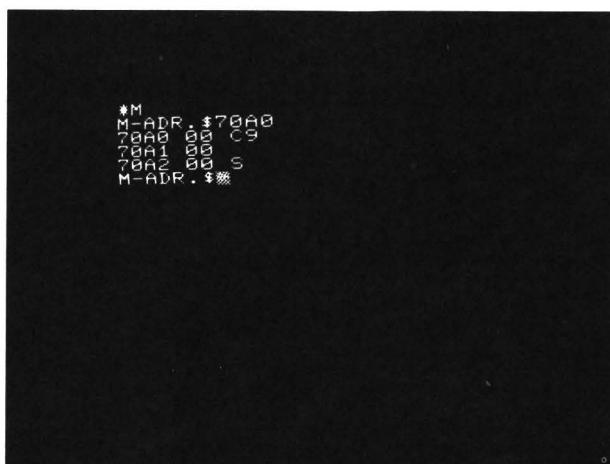



図 1—7

途中でメモリ・コレクションを行うアドレスを変える場合このようにして行うことができます。ここで、＄70A0 番地のデータが書き変えられたかどうか確かめるため、同じアドレスを与えてみます。図1－8に示すように＄70A0 上のデータは＄C9になっていることが確かめられます。[†)]

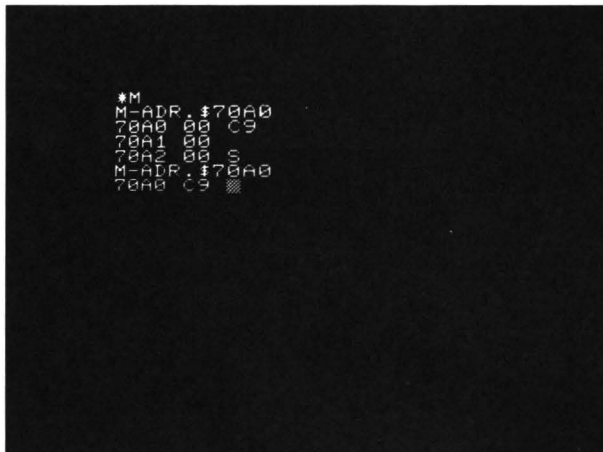



図 1—8

Mコマンドをキャンセルしてモニタ・コマンドレベルへ戻るには、BREAK キーを押します。

†) メモリ・コレクションを行ったあとでは、データが正しくセットされたか今のように確かめる習慣をつけるように。機械語プログラムの実行は、コードデータをオペコードとしてそのまま実行してしまい、コードが1ビット異なれば全然別の働きをしてしまうからです。大ていの場合、それはプログラムの暴走、破壊を引き起こします。なお、データの確認は、次のDコマンドでまとめて行うこともできます。

1.2.2 Dコマンド（D：memory dump）

機　能　指定するメモリ・ブロックの内容をまとめて表示します。

解　説　Dコマンドを与えると、図1－9の表示がなされます。

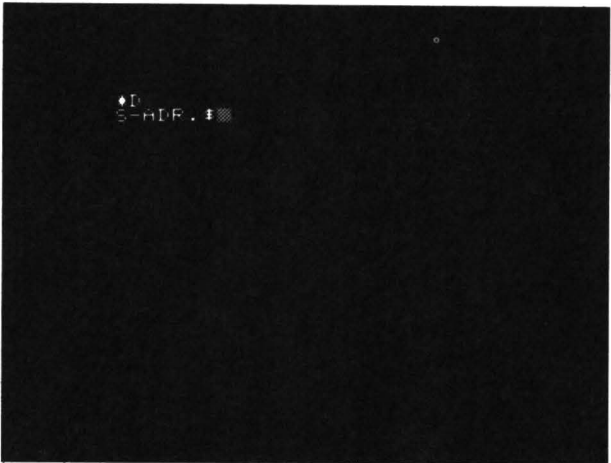



図 1−9

ここでモニタは、メモリ・ダンプを行うメモリ・ブロックの先頭アドレスを要求しています。たとえば、メモリ・ブロック＄0000～＄007Fまで（これはモニタ・プログラム自身の一部である）をダンプするには、S-ADR 即ちスタートアドレスとして"＄"記号につづけて"0000"をキー入力し CR キーを押します。すると図1－10に示す表示がなされます。

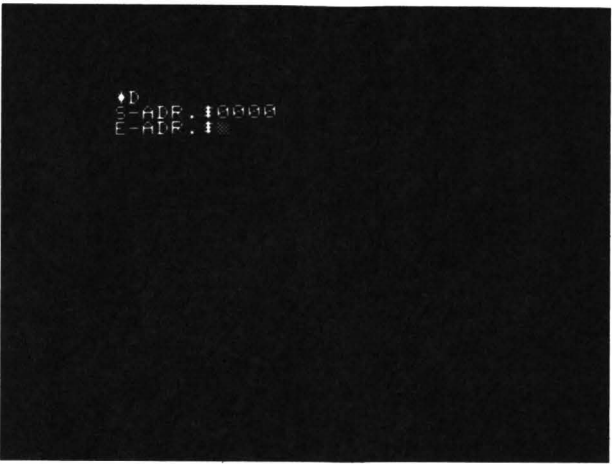



図 1−10

ここでE-ADRは、メモリ・ダンプを行うメモリ・ブロックのエンドアドレスなので、"＄"記号につづけて"007F"をキー入力し［CR］キーを押すと、図1－11のように、このブロックの内容が表示されます。

図　1－11

このように、1行に8バイトずつまとめて表示されます。

メモリ・ダンプの実行中にスペースバーを押すと、押し続ける間、表示はストップしているので、大きなメモリ・ブロックをダンプする場合、時々止めながらデータを確認することができます。

1.2.3 Jコマンド（J：jump）

機　能　指定するアドレスへコントロールを移します。正確にいうと、指定するアドレスを、CPU のプログラムカウンタに与えることになります。

解　説　Jコマンドを与えると、図1－12に示す表示がなされます。

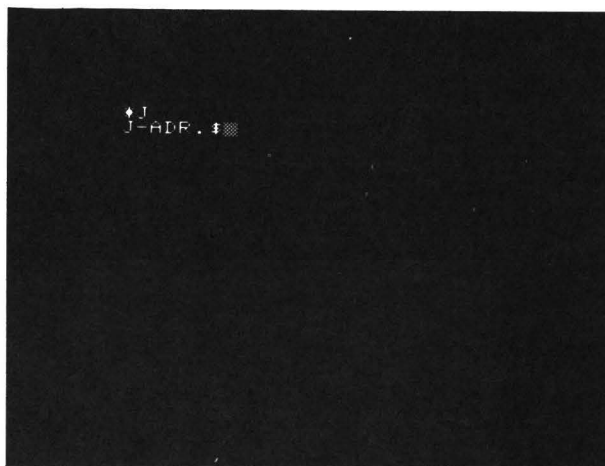



図　1－12

ここでモニタは、ジャンプすべきアドレス指定を要求しており、前のコマンドと同様に"＄"記号につづけて、実行したい機械語プログラムの入口のアドレスを16進4桁でキー入力し、［CR］キーを押します。すると、Jコマンドからもモニタからも、コントロールは離れることになり、その後のコンピュータの動作は、ジャンプした先の機械語プログラム次第となります。(不注意にJコマンドを実行させると、システムの暴走を招くことになる可能性があります。†))
プログラム実行中にオペコード＄FF（RST 7命令）に遭遇し実行すると、モニタは、AF、BC、DE、HL、および＄FFコードの置かれているアドレスの値をCRTディスプレイに表示し、コントロールをモニタコマンドレベルへ戻します。従って＄FFコードを、デバッグを行う際のブレークポイント（break point）として使うことができます。
BASIC インタープリタなどの、もとのシステムプログラムを壊していない場合は、このコマンドで、それらへ復帰することができます。復帰のし方には、HOT スタートと COLD スタートがあります。HOT スタートとは、システムプログラムのワークエリアとプログラムエリアを、そのままにしてスタートするもの——つまり、モニタへ移る前の状態が生きていて hot という意味です——であり、COLD スタートは、IPL による起動時と同じように、初期状態——冷めたい状態——にしてスタートするものです。
BASIC インタープリタのそれぞれのスタートアドレスは次のようになっています。

HOT　スタートアドレス＝＄1300
COLD　スタートアドレス＝＄12A0

---

†）ハードウェアを壊すことはありませんが、RAM上のプログラムの破壊や、書き込み禁止の筈のテープに何かを書き込んでしまうというようなことが起こり得ます。

1.2.4 Sコマンド（S：save）

機　能　指定するメモリブロックの内容を、ファイルネームを付けてカセットテープ上に記録します。

解　説　Sコマンドを与えると、図1－13に示す表示がなされます。

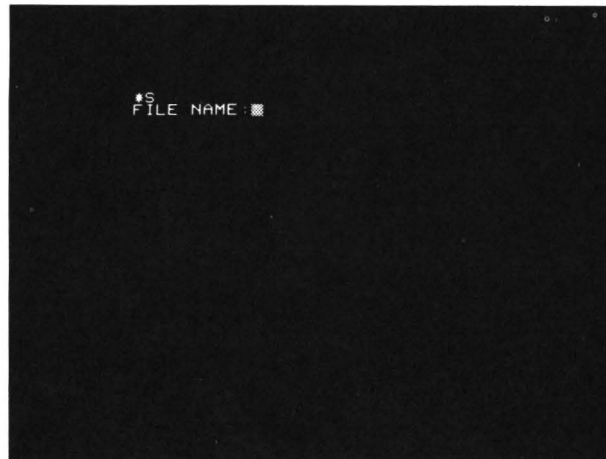



図　1－13

ここでモニタは、作成する機械語プログラムファイルのファイル名指定を要求しています。そこで、適当なファイル名を16文字以内でカーソル位置からキー入力して［CR］キーを押します。16文字を越えた分は無効となります。

たとえば、"ABRACADABRA"という名前を指定する†）と、図1－14のような表示が続きます。

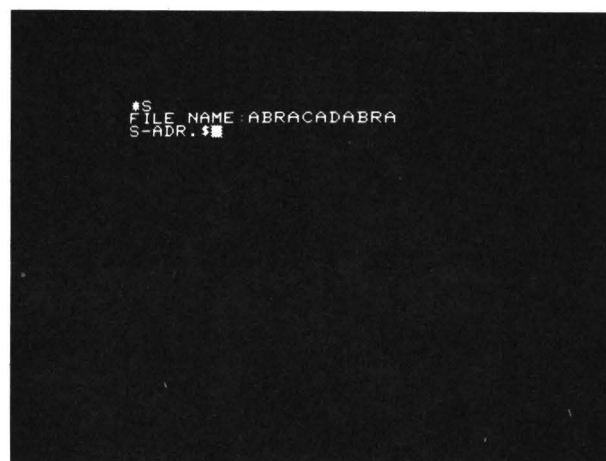



図　1－14

ファイル名を指定すると、モニタは次にセーブすべきメモリブロックの指定を要求するので、Dコマンドでメモリダンプすべきメモリブロックを指定するのと同様に S-ADR. および E-ADR. のそれぞれにスタートアドレスとエンドアドレスを指定します。

スタートアドレスもエンドアドレスも、メモリ上の、どこであっても、プロテクトされませんが、

†）もしファイル名を指定せずに［CR］キーを押せば、ファイル名なしということになり、これは好ましいことではありません。訳のわからないおまじないのようなファイル名を付けるのも考えものですが……。

注意しなくてはならないのは、モニタエリア内を含むと、ローディングが不可能なファイルが出来てしまうことがあることです。即ち、モニタ自身が働きながら、自分自身をセーブするとなると、チェックコード（sum check code）が合わなくなるからで、もし、モニタ自身を含むメモリ・ブロックを正しくセーブしたい時は、モニタを含むエリア全体を別のエリアへブロック転送してから行ってください。

ABRACADABRA とファイル名をつけて記録するプログラムが、たとえばメモリ・ブロック＄6000～＄60A3までにあるとすれば、その2つのアドレスを指定します。すると図1－15に示される表示がなされます。

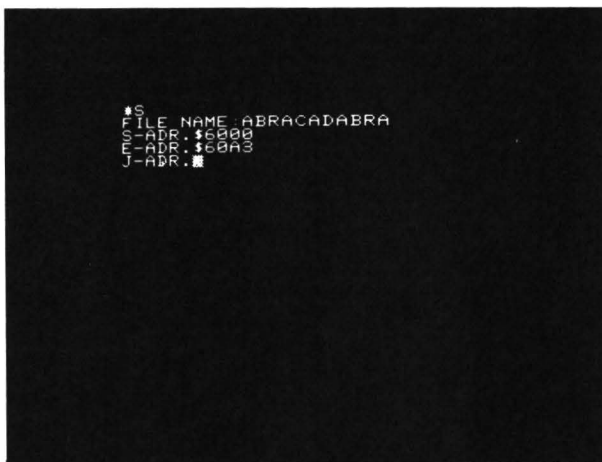



図 1－15

ここでモニタは、作成する機械語プログラムファイルのジャンプアドレスの指定を要求しており、ジャンプアドレスを指定しておくと、モニタコマンドL（load）によって、ローディングをした場合、ローディングの後に、続けてジャンプアドレスに指定されたアドレスへのジャンプを実行することになります。その機械語プログラムが独立したプログラムである場合、モニタのJコマンドが省けることになります。

一方、BASICプログラムとリンクして使う機械語サブルーチンプログラムなどは、ジャンプアドレスは指定しておきません。†) この場合、Lコマンドでローディングを行ったあとは、モニタコマンド待ちとなります。

例の"ABRACADABRA"プログラムは、ローディング後、＄6050番地から実行させたいプログラムだとすれば図1－16のようにジャンプアドレスを指定すればよいことになります。

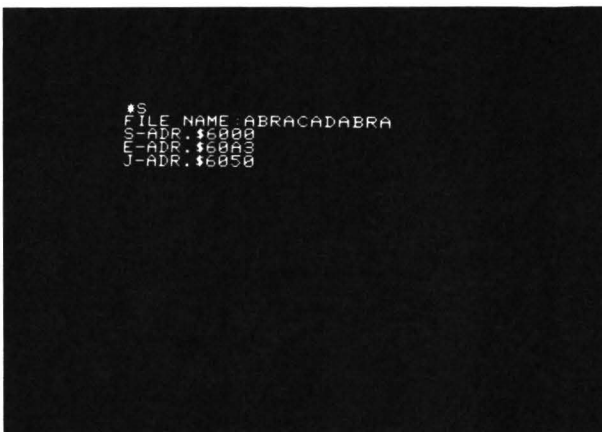



図 1－16

†) 単に CR キーを押します。

[CR] キーを押すと、自動的にセーブが行われます。
カセットテープが、カセットテープデッキにセットされない場合、カセットデッキのふたが開き"SET　TAPE"の表示がなされること、或いはライトプロテクトのテープがセットされているとき"WRITE　PROTECT"の表示がなされることは、 BASICコマンドSAVEの場合と同じです。

1.2.5　Vコマンド（V：verify）

機　能　カセットテープにセーブされたデータが、そのもとのメモリ・ブロック上のデータと同じであるかどうか調べます。

解　説　Vコマンドを与えると図1－17に示す表示が現われます。

図 1 －17

ここで、モニタは、ベリファイ（比較）すべき機械語プログラムファイルのファイル名の指定を要求しています。たとえば、Sコマンドの説明で作った、" ABRACADABRA " というファイルを比較するのであれば、図1－18に示すようにそのファイル名を指定します。——ただしあらかじめテープは巻き戻しておかなくてはなりません。

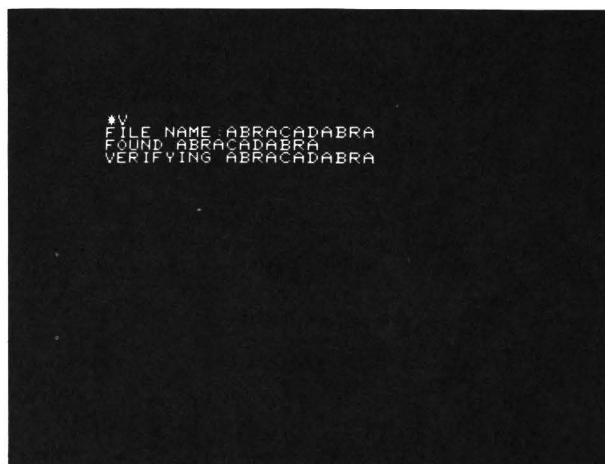



図 1－18

CR キーを押すと自動的にベリファイが実行されます。ベリファイすべきメモリ・ブロックは、ファイル上にファイルインフォメーションとして記録されているものによっています。

もし、ファイル名を指定しないと、最初に見つかった機械語プログラムファイルについてベリファイを実行することになります。

ファイルデータとメモリ・ブロック上のデータが同一であれば" OK " が表示され、異れば " ERROR " が表示されます。

Sコマンドを実行したら、つづけてこのVコマンドを実行しておくように心がけることをお勧めします。

1.2.6 Lコマンド (L：load)

機　能　指定するファイル名をもつ機械語プログラムファイルのローディングを行います。

解　説　Lコマンドを与えます。すると図1－19に示す表示が行われます。

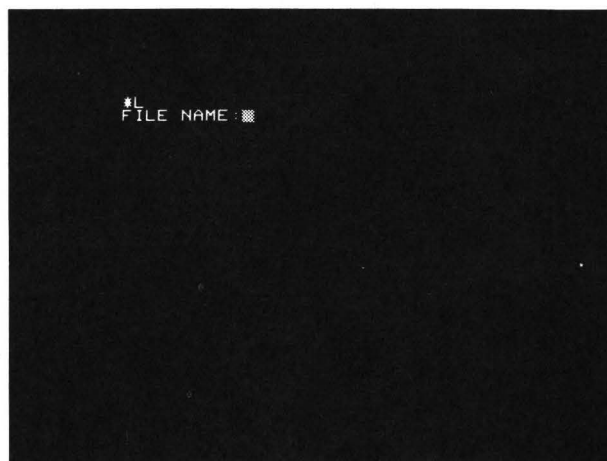



図　1－19

ここでモニタはロードすべき機械語 プログラムファイルのファイル名の指定を要求しているのでたとえば、Sコマンドの説明で作成した" ABRACADABRA "をロードするには図1－20のようにファイル名を指定します。

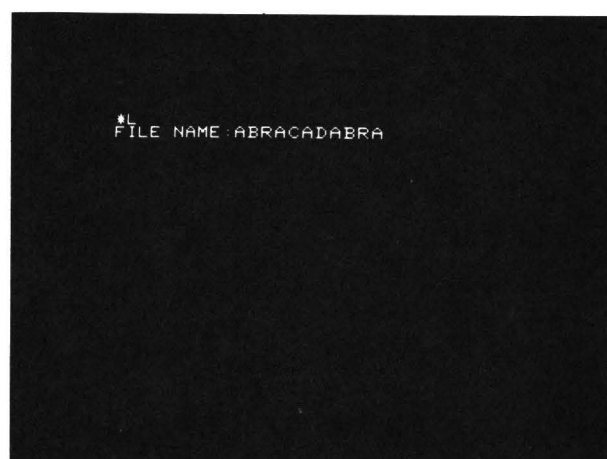



図　1－20

CR キーを押すと、まずファイル・サーチを行います。即ち、指定されたファイルを見つけ出し、見つかったらシステムへのローディングを行います。
図1－21は、ファイル名"OPEN SESAME"の次に "ABRACADABRA" が見つかり、ローディングを行っている状態を示しています。

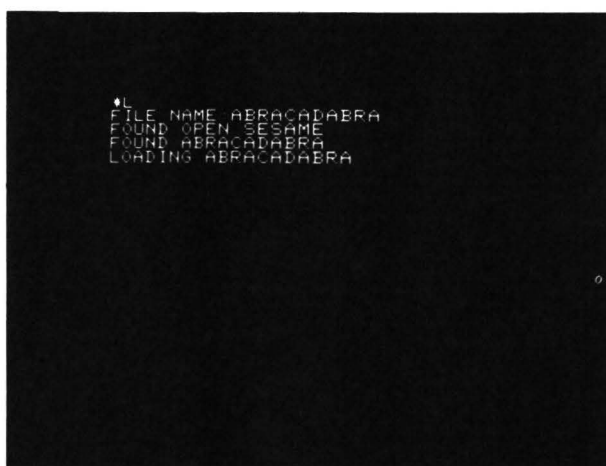



図 1－21

ローディングを行う場所は、Sコマンドでメモリ・ブロックを指定した同じメモリ・エリアになります。
プログラム" ABRACADABRA "の場合は、もともと＄6000～＄60A3のメモリ・ブロックの内容をSコマンドでファイルに記録したので、Lコマンドでロードすると図1－22に示すようにやはり、＄6000～＄60A3のメモリ・エリアにローディングされます。

図 1－22

図1－22の右端に"jump"と書いて、＄6050番地へ矢印を描いています。即ち、Sコマンドで"ABRACADABRA"を作成したとき、ジャンプアドレスとして、この番地を指定しておいたので、ローディングの終了後に、つづけて" ABRACADABRA "プログラムを＄6050番地から実行するということを示しています。もしジャンプアドレスを指定していなかったら、ローディング終了後は、モニタ・コマンド・レベルに戻ることになります。

1.3 モニタ・サブルーチン

モニタMZ-1Z001Mのサブルーチンを表１－１に示します。サブルーチン名は、本書の付録A.4のモニタ・プログラム・アセンブリ・リスト上にあるラベルです。このラベルは、それぞれのサブルーチンの機能をシンボリックに表現しています。

レジスタ保存は、そのサブルーチンをコールし、リターンした時に、内容が保存されるレジスタを示しています。即ち、それ以外のレジスタの内容はリターン時に一般に変化することになるので、注意しなくてはなりません。

スタック数は、そのサブルーチンをコールした時使用される最大のスタックの重なりの数です。たとえば、サブルーチンの"LETLN"はスタック数が８なので、スタックエリア８×２＝16バイトが必要なわけです。スタックエリアはプログラマが必要サイズ確保し、プログラムエリアや他のワークエリアを壊さないように管理してやらなければなりません。

サブルーチンを呼び出すには、機械語命令CALL または、BASICのUSR関数を用います。

即ち、一般に

CALL　subroutine address
USR　($ subroutine address)
subroutine address：16進４桁

の形の命令を実行します。

たとえば、モニタサブルーチン LETNL をコールして改行 (let new line) を実行させるには、CALL 0A2EHまたはUSR（＄0A2E)、即ち機械語命令、

CD2E0A　(hex)

を実行してやればよいことになります。

表　1－1　モニタ・サブルーチン

サブルーチン名とアドレス（16進アドレス）	サブルーチンの機能	スタック数
CALL LETNL **＄0A2E**	行を替えて次の行の先頭にカーソルを移動させます。 AF 以外のレジスタの内容は保存します。	8
CALL PRNTS **$08C4**	ＣＲＴディスプレイのカーソル位置にスペースを１個だけ表示します。カーソルは１文字右へ移動します。	3
CALL PRNT **$08C6**	Ａレジスタ（アキュムレータ）内のASCII コードデータに相当するキャラクタを、ＣＲＴディスプレイのカーソル位置に表示します。 即ち、ASCII コード＄１Ｅ～＄FF については、キャラクタを表示するサブルーチンとなります。 一方、ASCII コード＄00～＄1Ｄは、コントロールコードになっているので、それぞれディスプレイコントロールが実行されます。たとえば、Ａレジスタの内容が＄01であれば [↓] キーと同じくカーソルダウンが実行されます。	3

サブルーチン名とアドレス（16進アドレス）	サブルーチンの機能	スタック数
CALL MSG ＄0889	DEレジスタの示すアドレスからはじまるACCIIコードを続けてCRTディスプレイのカーソル位置から表示します。データのエンドマークは＄０D（キャリッジリターンコード）です。ただしこの場合、キャリッジリターンは実行されません。ASCCIIコード＄00～＄１Dについては、ディスプレイコントロールが実行されます。 全レジスタの内容が保存されます。	4
CALL BELL ＄0F14	中音のラ（約440Hz）をピッと鳴らします。	4
CALL MELDY ＄0F3F	音楽データを演奏します。音楽データの先頭アドレスは、あらかじめDEレジスタに指定しておきます。音楽データは「BASIC編」のMUSICステートメントに解説したデータと同様に、音程音長の順にASCIIコードで並べます。エンドマークは＄０D（キャリッジリターンコード）または＄２A（キャラクタ"＊"）でなければなりません。ただしリターン時にCフラグが，０なら演奏完了、１なら途中 BREAK キーが押されたことを示します。	
CALL XTEMP ＄0E50	演奏テンポを設定します。テンポデータ（＄01～＄07）をAレジスタにセットしてコールします。 Aレジスタ←―＄01：最も遅いテンポ Aレジスタ←―＄04：中位のテンポ Aレジスタ←―＄07：最も早いテンポ	3
CALL SOUT ＄0F22	任意の音程、音長の音を発生します。BCレジスタに音長を、HLレジスタに音程をセットしておいてコールします。（参考：HLレジスタに音程データ＄00A４をセットすると中音のラの音程になります。）	3
CALL TIMST ＄0E5E	内蔵時計に時刻をセットします。時刻の設定は、 Aレジスタ←―０：AM　　Aレジスタ←―１：PM DEレジスタ←―秒単位の時間（２バイトデータ）	3
CALL TIMRD ＄0EA9	内蔵時計の現在時刻を読み取ります。リターン時には、 Aレジスタ←―０：AM　　Aレジスタ←―１：PM DEレジスタ←―秒単位の時間（２バイトデータ） がセットされます。	3
CALL BRKEY ＄0562	BREAK キーが押されているかどうかチェックします。 Zフラグ←―０：押されていない Zフラグ←―１：押されている	2
CALL GETL ＄06A4	キーボードから１行のデータを入力します。入力データのストアされる先頭アドレスと、入力可能な文字数は次のように設定しなければなりません。 DEレジスタ←―入力データをストアする先頭アドレス GETL－２番地（＄06A２）←―入力可能な文字数の最大 キー入力は CR キー（または ENT キー）を押すことにより終わりま	8

サブルーチン名とアドレス(16進アドレス)	サブルーチン機能	スタック数
	す。その時、エンドマーク＄０Ｄが入力データに続いてストアされます。従って、入力可能な文字数には、このエンドマーク１文字分も含めておかなければなりません。 キー入力の際、CRTディスプレイ上にエコーバックが行われ、カーソルコントロール、挿入、削除なども受け付けられます。 途中で BREAK キーを押すと、ＤＥレジスタの示す先頭アドレスにブレークコード＄０Ｂがセットされ、リターンします。 モニタ・プログラム自身も、このサブルーチンを使用しています。そしてＤＥレジスタにラベル"BUFER"（＄1093番地）をセットし、GETL-2番地に＄27をセットして使用しています。	
CALL GETKY ＄0832	キーボードから１文字だけデータをＡレジスタに取り込みます。たとえば B キーを押しているとき、CALL GETKY を行うと、Ａレジスタにキャラクタ"Ｂ"のASCII コード＄42がセットされリターンします。キーが押されていないとＡレジスタの内容は＄00でリターンします。 キー入力によるチャタリングは防止しておらず、またCRTディスプレイ上へのエコーバックも行われません。	8
CALL ASC ＄05F3	Ａレジスタの下位４ビット（＄０～＄Ｆ）を16進数を表現するキャラクタ（"０"～"Ｆ"）の ASCII コードに変換し、それをＡレジスタにセットします。 たとえば、Ａレジスタの下位４ビットが＄８（２進数の1000）であったとすると、リターン時には、Ａレジスタに＄38（キャラクタ"８"のASCII コード）がセットされます。	1
CALL HEX ＄05FD	サブルーチンASCの逆で、Ａレジスタの内容が16進数を示すキャラクタのASCII コードである場合、その16進数をＡレジスタの下位４ビットにセットしてリターンします。 たとえば、Ａレジスタの内容が＄42、即ちキャラクタ"Ｂ"のASCIIコードである場合、リターン時にＡレジスタには＄０Ｂがセットされています。 ただし、Ａレジスタの内容が、16進数を表現するASCIIコード以外であれば、Ｃフラグ（carry flag）が１となり、Ａレジスタの内容は、不定の値となります。 リターン時のＣフラグ←─０：Ａレジスタ下位４ビット＝16進数 リターン時のＣフラグ←─１：データエラー	1
CALL HLHEX ＄0614	連続した４バイトのデータが４桁の16進数を表現するキャラクタのASCII コード列であるなら、その16進数をＨＬレジスタにセットしてリターンします。 該当する４バイトデータが16進数を表現するキャラクタのASCIIコード列でないなら、Ｃフラグが１となりＨＬレジスタの内容は不定です。 ＤＥレジスタ←─連続した４バイトデータの先頭アドレスをセットする。 リターン時のＣフラグ←─０：HLレジスタ＝２桁の16進数 リターン時のＣフラグ←─１：データエラー たとえば、＄8000番地から、データ＄33、＄30、＄43、＄39（即ちキャラク	4

サブルーチン名とアドレス(16進アドレス)	サブルーチン機能	スタック数
	タ"30C 9")がある場合。 LD DE、8000H (H:hexadecimal) CALL 0614H を実行すると、リターン時にHLレジスタの内容は$30C 9となります。	
CALL 2HEX $0623	連続した2バイトのデータが、2桁の16進数を表現するキャラクタのASCII コード列であるなら、その16進数をAレジスタにセットしてリターンします。 該当する2バイトのデータが、16進数を表現するキャラクタのASCIIコード列でないなら、Cフラグが1となり、Aレジスタの内容は不定です。	2
CALL ?PONT $0C29	現在のカーソル位置、即ち、Video-RAM上の該当アドレスをHLレジスタにセットしてリターンします。 たとえば、カーソルがホーム位置(左上隅)に位置しているなら、リターン時のHLレジスタの内容は$D000となります。	2

付　録　Appendix

次の５つの付録がここに置かれています。

■ Z80A CPUの全インストラクションのニーモニックコードとオペコードの対照表Ａ・1とＡ・2

Ａ・1はニーモニックコードをアルファベット順に並べており、プログラマがZ80A CPUの或るインストラクションのオブジェクトコード（16進表記機械語コード）を知りたい時に利用します。

Ａ・2はオペコードの16進順に並べてあり、逆にオブジェクトコードからインストラクションを調べたいとき利用してください。

各インストラクションのオペレーション、フラグオペレーション、実行時間などの詳細については、MZ-2000 OWNER'S MANUALの付録にあるZ80A CPUテクニカルデータを参照してください。

■ IPLプログラム・アセンブリ・リスト：A3

プログラマの参考のため、IPLプログラムの全アセンブリ・リストを示しています。

■ MONITOR MZ-1Z001Mのプログラム・リスト：Ａ・4

プログラマの参考のため、MONITOR MZ-1Z001Mの全アセンブリ・リストを示しています。

■ BASICテープのコピー作成と改訂の方法：Ａ・5

A.1 Z80A-CPUインストラクションセット対照コード表1
ニーモニックコードのアルファベット順

注

ニーモニックコードのオペランドにあるnn, n, d, eという記号はプログラマが指定する定数データを表わしています。オペランド欄には、例として、これらの定数データがそれぞれ次の値である場合のコードを示しています。

nn ＝ 584H
n ＝ 20H
d ＝ 5
e ＝ 30H

そして例の値であることを明示するためにデータコードをイタリック体とし、さらに下線を施しています。

オペコード	ニーモニックコード
8E	ADC　A, (HL)
DD8E*05*	ADC　A, (IX+d)
FD8E*05*	ADC　A, (IY+d)
8F	ADC　A, A
88	ADC　A, B
89	ADC　A, C
8A	ADC　A, D
8B	ADC　A, E
8C	ADC　A, H
8D	ADC　A, L
CE*20*	ADC　A, n
ED4A	ADC　HL, BC
ED5A	ADC　HL, DE
ED6A	ADC　HL, HL
ED7A	ADC　HL, SP
86	ADD　A, (HL)
DD86*05*	ADD　A, (IX+d)
FD86*05*	ADD　A, (IY+d)
87	ADD　A, A
80	ADD　A, B
81	ADD　A, C
82	ADD　A, D
83	ADD　A, E
84	ADD　A, H
85	ADD　A, L
C6*20*	ADD　A, n
09	ADD　HL, BC
19	ADD　HL, DE
29	ADD　HL, HL
39	ADD　HL, SP
DD09	ADD　IX, BC
DD19	ADD　IX, DE
DD29	ADD　IX, IX
DD39	ADD　IX, SP
FD09	ADD　IY, BC
FD19	ADD　IY, DE
FD29	ADD　IY, IY
FD39	ADD　IY, SP

オペコード	ニーモニックコード
A6	AND (HL)
DDA605	AND (IX+d)
FDA605	AND (IY+d)
A7	AND A
A0	AND B
A1	AND C
A2	AND D
A3	AND E
A4	AND H
A5	AND L
E620	AND n
CB46	BIT 0, (HL)
DDCB0546	BIT 0, (IX+d)
FDCB0546	BIT 0, (IY+d)
CB47	BIT 0, A
CB40	BIT 0, B
CB41	BIT 0, C
CB42	BIT 0, D
CB43	BIT 0, E
CB44	BIT 0, H
CB45	BIT 0, L
CB4E	BIT 1, (HL)
DDCB054E	BIT 1, (IX+d)
FDCB054E	BIT 1, (IY+d)
CB4F	BIT 1, A
CB48	BIT 1, B
CB49	BIT 1, C
CB4A	BIT 1, D
CB4B	BIT 1, E
CB4C	BIT 1, H
CB4D	BIT 1, L
CB56	BIT 2, (HL)
DDCB0556	BIT 2, (IX+d)
FDCB0556	BIT 2, (IY+d)
CB57	BIT 2, A
CB50	BIT 2, B
CB51	BIT 2, C
CB52	BIT 2, D
CB53	BIT 2, E

オペコード	ニーモニックコード
CB54	BIT 2, H
CB55	BIT 2, L
CB5E	BIT 3, (HL)
DDCB055E	BIT 3, (IX+d)
FDCB055E	BIT 3, (IY+d)
CB5F	BIT 3, A
CB58	BIT 3, B
CB59	BIT 3, C
CB5A	BIT 3, D
CB5B	BIT 3, E
CB5C	BIT 3, H
CB5D	BIT 3, L
CB66	BIT 4, (HL)
DDCB0566	BIT 4, (IX+d)
FDCB0566	BIT 4, (IY+d)
CB67	BIT 4, A
CB60	BIT 4, B
CB61	BIT 4, C
CB62	BIT 4, D
CB63	BIT 4, E
CB64	BIT 4, H
CB65	BIT 4, L
CB6E	BIT 5, (HL)
DDCB056E	BIT 5, (IX+d)
FDCB056E	BIT 5, (IY+d)
CB6F	BIT 5, A
CB68	BIT 5, B
CB69	BIT 5, C
CB6A	BIT 5, D
CB6B	BIT 5, E
CB6C	BIT 5, H
CB6D	BIT 5, L
CB76	BIT 6, (HL)
DDCB0576	BIT 6, (IX+d)
FDCB0576	BIT 6, (IY+d)
CB77	BIT 6, A
CB70	BIT 6, B
CB71	BIT 6, C
CB72	BIT 6, D
CB73	BIT 6, E

オペコード	ニーモニックコード
CB74	BIT 6,H
CB75	BIT 6,L
CB7E	BIT 7,(HL)
DDCB057E	BIT 7,(IX+d)
FDCB057E	BIT 7,(IY+d)
CB7F	BIT 7,A
CB78	BIT 7,B
CB79	BIT 7,C
CB7A	BIT 7,D
CB7B	BIT 7,E
CB7C	BIT 7,H
CB7D	BIT 7,L
DC8405	CALL C,nn
FC8405	CALL M,nn
D48405	CALL NC,nn
CD8405	CALL nn
C48405	CALL NZ,nn
F48405	CALL P,nn
EC8405	CALL PE,nn
E48405	CALL PO,nn
CC8405	CALL Z,nn
3F	CCF
BE	CP (HL)
DDBE05	CP (IX+d)
FDBE05	CP (IY+d)
BF	CP A
B8	CP B
B9	CP C
BA	CP D
BB	CP E
BC	CP H
BD	CP L
FE20	CP n
EDA9	CPD
EDB9	CPDR
EDA1	CPI

オペコード	ニーモニックコード
EDB1	CPIR
2F	CPL
27	DAA
35	DEC (HL)
DD3505	DEC (IX+d)
FD3505	DEC (IY+d)
3D	DEC A
05	DEC B
0B	DEC BC
0D	DEC C
15	DEC D
1B	DEC DE
1D	DEC E
25	DEC H
2B	DEC HL
DD2B	DEC IX
FD2B	DEC IY
2D	DEC L
3B	DEC SP
F3	DI
102E	DJNZ e
FB	EI
E3	EX (SP),HL
DDE3	EX (SP),IX
FDE3	EX (SP),IY
08	EX AF,AF’
EB	EX DE,HL
D9	EXX
76	HALT
ED46	IM 0
ED56	IM 1

オペコード	ニーモニックコード
ED5E	IM 2
ED78	IN A,(C)
DB*20*	IN A,(n)
ED40	IN B,(C)
ED48	IN C,(C)
ED50	IN D,(C)
ED58	IN E,(C)
ED60	IN H,(C)
ED68	IN L,(C)
34	INC (HL)
DD34*05*	INC (IX+d)
FD34*05*	INC (IY+d)
3C	INC A
04	INC B
03	INC BC
0C	INC C
14	INC D
13	INC DE
1C	INC E
24	INC H
23	INC HL
DD23	INC IX
FD23	INC IY
2C	INC L
33	INC SP
EDAA	IND
EDBA	INDR
EDA2	INI
EDB2	INIR
E9	JP (HL)
DDE9	JP (IX)
FDE9	JP (IY)
DA*8405*	JP C,nn
FA*8405*	JP M,nn
D2*8405*	JP NC,nn
C3*8405*	JP nn

オペコード	ニーモニックコード
C2*8405*	JP NZ,nn
F2*8405*	JP P,nn
EA*8405*	JP PE,nn
E2*8405*	JP PO,nn
CA*8405*	JP Z,nn
38*2E*	JR C,e
18*2E*	JR e
30*2E*	JR NC,e
20*2E*	JR NZ,e
28*2E*	JR Z,e
02	LD (BC),A
12	LD (DE),A
77	LD (HL),A
70	LD (HL),B
71	LD (HL),C
72	LD (HL),D
73	LD (HL),E
74	LD (HL),H
75	LD (HL),L
36*20*	LD (HL),n
DD77*05*	LD (IX+d),A
DD70*05*	LD (IX+d),B
DD71*05*	LD (IX+d),C
DD72*05*	LD (IX+d),D
DD73*05*	LD (IX+d),E
DD74*05*	LD (IX+d),H
DD75*05*	LD (IX+d),L
DD36*0520*	LD (IX+d),n
FD77*05*	LD (IY+d),A
FD70*05*	LD (IY+d),B
FD71*05*	LD (IY+d),C
FD72*05*	LD (IY+d),D
FD73*05*	LD (IY+d),E
FD74*05*	LD (IY+d),H
FD75*05*	LD (IY+d),L
FD36*0520*	LD (IY+d),n
32*8405*	LD (nn),A
ED43*8405*	LD (nn),BC

オペコード	ニーモニックコード
ED53*8405*	LD (nn), DE
22*8405*	LD (nn), HL
DD22*8405*	LD (nn), IX
FD22*8405*	LD (nn), IY
ED73*8405*	LD (nn), SP
0A	LD A, (BC)
1A	LD A, (DE)
7E	LD A, (HL)
DD7E*05*	LD A, (IX+d)
FD7E*05*	LD A, (IY+d)
3A*8405*	LD A, (nn)
7F	LD A, A
78	LD A, B
79	LD A, C
7A	LD A, D
7B	LD A, E
7C	LD A, H
ED57	LD A, I
7D	LD A, L
3E*20*	LD A, n
46	LD B, (HL)
DD46*05*	LD B, (IX+d)
FD46*05*	LD B, (IY+d)
47	LD B, A
40	LD B, B
41	LD B, C
42	LD B, D
43	LD B, E
44	LD B, H
45	LD B, L
06*20*	LD B, n
ED4B*8405*	LD BC, (nn)
01*8405*	LD BC, nn
4E	LD C, (HL)
DD4E*05*	LD C, (IX+d)
FD4E*05*	LD C, (IY+d)
4F	LD C, A
48	LD C, B
49	LD C, C
4A	LD C, D

オペコード	ニーモニックコード
4B	LD C, E
4C	LD C, H
4D	LD C, L
0E*20*	LD C, n
56	LD D, (HL)
DD56*05*	LD D, (IX+d)
FD56*05*	LD D, (IY+d)
57	LD D, A
50	LD D, B
51	LD D, C
52	LD D, D
53	LD D, E
54	LD D, H
55	LD D, L
16*20*	LD D, n
ED5B*8405*	LD DE, (nn)
11*8405*	LD DE, nn
5E	LD E, (HL)
DD5E*05*	LD E, (IX+d)
FD5E*05*	LD E, (IY+d)
5F	LD E, A
58	LD E, B
59	LD E, C
5A	LD E, D
5B	LD E, E
5C	LD E, H
5D	LD E, L
1E*20*	LD E, n
66	LD H, (HL)
DD66*05*	LD H, (IX+d)
FD66*05*	LD H, (IY+d)
67	LD H, A
60	LD H, B
61	LD H, C
62	LD H, D
63	LD H, E
64	LD H, H
65	LD H, L
26*20*	LD H, n
2A*8405*	LD H, (nn)

オペコード	ニーモニックコード
21*8405*	LD HL,nn
ED47	LD I,A
DD2A*8405*	LD IX,(nn)
DD21*8405*	LD IX,nn
FD2A*8405*	LD IY,(nn)
FD21*8405*	LD IY,nn
6E	LD L,(HL)
DD6E*05*	LD L,(IX+d)
FD6E*05*	LD L,(IY+d)
6F	LD L,A
68	LD L,B
69	LD L,C
6A	LD L,D
6B	LD L,E
6C	LD L,H
6D	LD L,L
2E*20*	LD L,n
ED7B*8405*	LD SP,(nn)
F9	LD SP,HL
DDF9	LD SP,IX
FDF9	LD SP,IY
31*8405*	LD SP,nn
EDA8	LDD
EDB8	LDDR
EDA0	LDI
EDB0	LDIR
ED44	NEG
00	NOP
B6	OR (HL)
DDB6*05*	OR (IX+d)
FDB6*05*	OR (IY+d)
B7	OR A
B0	OR B
B1	OR C
B2	OR D
B3	OR E

オペコード	ニーモニックコード
B4	OR H
B5	OR L
F6*20*	OR n
EDBB	OTDR
EDB3	OTIR
ED79	OUT (C),A
ED41	OUT (C),B
ED49	OUT (C),C
ED51	OUT (C),D
ED59	OUT (C),E
ED61	OUT (C),H
ED69	OUT (C),L
D3*20*	OUT (n),A
EDAB	OUTD
EDA3	OUTI
F1	POP AF
C1	POP BC
D1	POP DE
E1	POP HL
DDE1	POP IX
FDE1	POP IY
F5	PUSH AF
C5	PUSH BC
D5	PUSH DE
E5	PUSH HL
DDE5	PUSH IX
FDE5	PUSH IY
CB86	RES 0,(HL)
DDCB*05*86	RES 0,(IX+d)
FDCB*05*86	RES 0,(IY+d)
CB87	RES 0,A
CB80	RES 0,B
CB81	RES 0,C
CB82	RES 0,D
CB83	RES 0,E
CB84	RES 0,H

オペコード	ニーモニックコード
CB85	RES 0,L
CB8E	RES 1,(HL)
DDCB*05*8E	RES 1,(IX+d)
FDCB*05*8E	RES 1,(IY+d)
CB8F	RES 1,A
CB88	RES 1,B
CB89	RES 1,C
CB8A	RES 1,D
CB8B	RES 1,E
CB8C	RES 1,H
CB8D	RES 1,L
CB96	RES 2,(HL)
DDCB*05*96	RES 2,(IX+d)
FDCB*05*96	RES 2,(IY+d)
CB97	RES 2,A
CB90	RES 2,B
CB91	RES 2,C
CB92	RES 2,D
CB93	RES 2,E
CB94	RES 2,H
CB95	RES 2,L
CB9E	RES 3,(HL)
DDCB*05*9E	RES 3,(IX+d)
FDCB*05*9E	RES 3,(IY+d)
CB9F	RES 3,A
CB98	RES 3,B
CB99	RES 3,C
CB9A	RES 3,D
CB9B	RES 3,E
CB9C	RES 3,H
CB9D	RES 3,L
CBA6	RES 4,(HL)
DDCB*05*A6	RES 4,(IX+d)
FDCB*05*A6	RES 4,(IY+d)
CBA7	RES 4,A
CBA0	RES 4,B
CBA1	RES 4,C
CBA2	RES 4,D
CBA3	RES 4,E
CBA4	RES 4,H

オペコード	ニーモニックコード
CBA5	RES 4,L
CBAE	RES 5,(HL)
DDCB*05*AE	RES 5,(IX+d)
FDCB*05*AE	RES 5,(IY+d)
CBAF	RES 5,A
CBA8	RES 5,B
CBA9	RES 5,C
CBAA	RES 5,D
CBAB	RES 5,E
CBAC	RES 5,H
CBAD	RES 5,L
CBB6	RES 6,(HL)
DDCB*05*B6	RES 6,(IX+d)
FDCB*05*B6	RES 6,(IY+d)
CBB7	RES 6,A
CBB0	RES 6,B
CBB1	RES 6,C
CBB2	RES 6,D
CBB3	RES 6,E
CBB4	RES 6,H
CBB5	RES 6,L
CBBE	RES 7,(HL)
DDCB*05*BE	RES 7,(IX+d)
FDCB*05*BE	RES 7,(IY+d)
CBBF	RES 7,A
CBB8	RES 7,B
CBB9	RES 7,C
CBBA	RES 7,D
CBBB	RES 7,E
CBBC	RES 7,H
CBBD	RES 7,L
C9	RET
D8	RET C
F8	RET M
D0	RET NC
C0	RET NZ
F0	RET P
E8	RET PE
E0	RET PO

オペコード	ニーモニックコード
C8	RET Z
ED4D	RETI
ED45	RETN
CB16	RL (HL)
DDCB*05*16	RL (IX+d)
FDCB*05*16	RL (IY+d)
CB17	RL A
CB10	RL B
CB11	RL C
CB12	RL D
CB13	RL E
CB14	RL H
CB15	RL L
17	RLA
CB06	RLC (HL)
DDCB*05*06	RLC (IX+d)
FDCB*05*06	RLC (IY+d)
CB07	RLC A
CB00	RLC B
CB01	RLC C
CB02	RLC D
CB03	RLC E
CB04	RLC H
CB05	RLC L
07	RLCA
ED6F	RLD
CB1E	RR (HL)
DDCB*05*1E	RR (IX+d)
FDCB*05*1E	RR (IY+d)
CB1F	RR A
CB18	RR B
CB19	RR C
CB1A	RR D
CB1B	RR E
CB1C	RR H
CB1D	RR L
1F	RRA

オペコード	ニーモニックコード
CB0E	RRC (HL)
DDCB*05*0E	RRC (IX+d)
FDCB*05*0E	RRC (IY+d)
CB0F	RRC A
CB08	RRC B
CB09	RRC C
CB0A	RRC D
CB0B	RRC E
CB0C	RRC H
CB0D	RRC L
0F	RRCA
ED67	RRD
C7	RST 00H
CF	RST 08H
D7	RST 10H
DF	RST 18H
E7	RST 20H
EF	RST 28H
F7	RST 30H
FF	RST 38H
9E	SBC A,(HL)
DD9E*05*	SBC A,(IX+d)
FD9E*05*	SBC A,(IY+d)
9F	SBC A,A
98	SBC A,B
99	SBC A,C
9A	SBC A,D
9B	SBC A,E
9C	SBC A,H
9D	SBC A,L
DE*20*	SBC A,n
ED42	SBC HL,BC
ED52	SBC HL,DE
ED62	SBC HL,HL
ED72	SBC HL,SP
37	SCF

オペコード	ニーモニックコード
CBC6	SET 0, (HL)
DDCB*05*C6	SET 0, (IX+d)
FDCB*05*C6	SET 0, (IY+d)
CBC7	SET 0, A
CBC0	SET 0, B
CBC1	SET 0, C
CBC2	SET 0, D
CBC3	SET 0, E
CBC4	SET 0, H
CBC5	SET 0, L
CBCE	SET 1, (HL)
DDCB*05*CE	SET 1, (IX+d)
FDCB*05*CE	SET 1, (IY+d)
CBCF	SET 1, A
CBC8	SET 1, B
CBC9	SET 1, C
CBCA	SET 1, D
CBCB	SET 1, E
CBCC	SET 1, H
CBCD	SET 1, L
CBD6	SET 2, (HL)
DDCB*05*D6	SET 2, (IX+d)
FDCB*05*D6	SET 2, (IY+d)
CBD7	SET 2, A
CBD0	SET 2, B
CBD1	SET 2, C
CBD2	SET 2, D
CBD3	SET 2, E
CBD4	SET 2, H
CBD5	SET 2, L
CBD8	SET 3, B
CBDE	SET 3, (HL)
DDCB*05*DE	SET 3, (IX+d)
FDCB*05*DE	SET 3, (IY+d)
CBDF	SET 3, A
CBD9	SET 3, C
CBDA	SET 3, D
CBDB	SET 3, E
CBDC	SET 3, H
CBDD	SET 3, L

オペコード	ニーモニックコード
CBE6	SET 4, (HL)
DDCB*05*E6	SET 4, (IX+d)
FDCB*05*E6	SET 4, (IY+d)
CBE7	SET 4, A
CBE0	SET 4, B
CBE1	SET 4, C
CBE2	SET 4, D
CBE3	SET 4, E
CBE4	SET 4, H
CBE5	SET 4, L
CBEE	SET 5, (HL)
DDCB*05*EE	SET 5, (IX+d)
FDCB*05*EE	SET 5, (IY+d)
CBEF	SET 5, A
CBE8	SET 5, B
CBE9	SET 5, C
CBEA	SET 5, D
CBEB	SET 5, E
CBEC	SET 5, H
CBED	SET 5, L
CBF6	SET 6, (HL)
DDCB*05*F6	SET 6, (IX+d)
FDCB*05*F6	SET 6, (IY+d)
CBF7	SET 6, A
CBF0	SET 6, B
CBF1	SET 6, C
CBF2	SET 6, D
CBF3	SET 6, E
CBF4	SET 6, H
CBF5	SET 6, L
CBFE	SET 7, (HL)
DDCB*05*FE	SET 7, (IX+d)
FDCB*05*FE	SET 7, (IY+d)
CBFF	SET 7, A
CBF8	SET 7, B
CBF9	SET 7, C
CBFA	SET 7, D
CBFB	SET 7, E
CBFC	SET 7, H
CBFD	SET 7, L

オペコード	ニーモニックコード
CB26	SLA (HL)
DDCB*05*26	SLA (IX+d)
FDCB*05*26	SLA (IY+d)
CB27	SLA A
CB20	SLA B
CB21	SLA C
CB22	SLA D
CB23	SLA E
CB24	SLA H
CB25	SLA L
CB2E	SRA (HL)
DDCB*05*2E	SRA (IX+d)
FDCB*05*2E	SRA (IY+d)
CB2F	SRA A
CB28	SRA B
CB29	SRA C
CB2A	SRA D
CB2B	SRA E
CB2C	SRA H
CB2D	SRA L
CB3E	SRL (HL)
DDCB*05*3E	SRL (IX+d)
FDCB*05*3E	SRL (IY+d)
CB3F	SRL A
CB38	SRL B
CB39	SRL C
CB3A	SRL D
CB3B	SRL E
CB3C	SRL H
CB3D	SRL L
96	SUB (HL)
DD96*05*	SUB (IX+d)
FD96*05*	SUB (IY+d)
97	SUB A
90	SUB B
91	SUB C
92	SUB D

オペコード	ニーモニックコード
93	SUB E
94	SUB H
95	SUB L
D6*20*	SUB n
AE	XOR (HL)
DDAE*05*	XOR (IX+d)
FDAE*05*	XOR (IY+d)
AF	XOR A
A8	XOR B
A9	XOR C
AA	XOR D
AB	XOR E
AC	XOR H
AD	XOR L
EE*20*	XOR n

A.2 Z80A-CPUインストラクションセット対照コード表2 オペコードの16進順

オペコード	ニーモニックコード
00	NOP
01*8405*	LD BC,nn
02	LD (BC),A
03	INC BC
04	INC B
05	DEC B
06*20*	LD B,n
07	RLCA
08	EX AF,AF'
09	ADD HL,BC
0A	LD A,(BC)
0B	DEC BC
0C	INC C
0D	DEC C
0E*20*	LD C,n
0F	RRCA
10*2E*	DJNZ e
11*8405*	LD DE,nn
12	LD (DE),A
13	INC DE
14	INC D
15	DEC D
16*20*	LD D,n
17	RLA
18*2E*	JR e
19	ADD HL,DE
1A	LD A,(DE)
1B	DEC DE
1C	INC E
1D	DEC E
1E*20*	LD E,n
1F	RRA
20*2E*	JR NZ,e
21*8405*	LD HL,nn
22*8405*	LD (nn),HL
23	INC HL
24	INC H
25	DEC H

注

ニーモニックコードのオペランドにあるnn, n, d, eという記号はプログラマが指定する定数データを表わしています。オペコード欄には、例として、これらの定数データがそれぞれ次の値である場合のコードを示しています。

nn ＝ 584H
n ＝ 20H
d ＝ 5
e ＝ 30H

そして例の値であることを明示するためにデータコードをイタリック体とし、さらに下線を施しています。

オペコードの1バイト目がCB, DD, EDまたはFDであるものについては表の後半にまとめられています。

オペコード	ニーモニックコード
26*20*	LD H,n
27	DAA
28*2E*	JR Z,e
29	ADD HL,HL
2A*8405*	LD HL,(nn)
2B	DEC HL
2C	INC L
2D	DEC L
2E*20*	LD L,n
2F	CPL
30*2E*	JR NC,e
31*8405*	LD SP,nn
32*8405*	LD (nn),A
33	INC SP
34	INC (HL)
35	DEC (HL)
36*20*	LD (HL),n
37	SCF
38*2E*	JR C,e
39	ADD HL,SP
3A*8405*	LD A,(nn)
3B	DEC SP
3C	INC A
3D	DEC A
3E*20*	LD A,n
3F	CCF
40	LD B,B
41	LD B,C
42	LD B,D
43	LD B,E
44	LD B,H
45	LD B,L
46	LD B,(HL)
47	LD B,A
48	LD C,B
49	LD C,C
4A	LD C,D
4B	LD C,E

オペコード	ニーモニックコード
4C	LD C,H
4D	LD C,L
4E	LD C,(HL)
4F	LD C,A
50	LD D,B
51	LD D,C
52	LD D,D
53	LD D,E
54	LD D,H
55	LD D,L
56	LD D,(HL)
57	LD D,A
58	LD E,B
59	LD E,C
5A	LD E,D
5B	LD E,E
5C	LD E,H
5D	LD E,L
5E	LD E,(HL)
5F	LD E,A
60	LD H,B
61	LD H,C
62	LD H,D
63	LD H,E
64	LD H,H
65	LD H,L
66	LD H,(HL)
67	LD H,A
68	LD L,B
69	LD L,C
6A	LD L,D
6B	LD L,E
6C	LD L,H
6D	LD L,L
6E	LD L,(HL)
6F	LD L,A
70	LD (HL),B

オペコード	ニーモニックコード
71	LD (HL),C
72	LD (HL),D
73	LD (HL),E
74	LD (HL),H
75	LD (HL),L
76	HALT
77	LD (HL),A
78	LD A,B
79	LD A,C
7A	LD A,D
7B	LD A,E
7C	LD A,H
7D	LD A,L
7E	LD A,(HL)
7F	LD A,A
80	ADD A,B
81	ADD A,C
82	ADD A,D
83	ADD A,E
84	ADD A,H
85	ADD A,L
86	ADD A,(HL)
87	ADD A,A
88	ADC A,B
89	ADC A,C
8A	ADC A,D
8B	ADC A,E
8C	ADC A,H
8D	ADC A,L
8E	ADC A,(HL)
8F	ADC A,A
90	SUB B
91	SUB C
92	SUB D
93	SUB E
94	SUB H
95	SUB L
96	SUB (HL)

オペコード	ニーモニックコード
97	SUB A
98	SBC A,B
99	SBC A,C
9A	SBC A,D
9B	SBC A,E
9C	SBC A,H
9D	SBC A,L
9E	SBC A,(HL)
9F	SBC A,A
A0	AND B
A1	AND C
A2	AND D
A3	AND E
A4	AND H
A5	AND L
A6	AND (HL)
A7	AND A
A8	XOR B
A9	XOR C
AA	XOR D
AB	XOR E
AC	XOR H
AD	XOR L
AE	XOR (HL)
AF	XOR A
B0	OR B
B1	OR C
B2	OR D
B3	OR E
B4	OR H
B5	OR L
B6	OR (HL)
B7	OR A
B8	CP B
B9	CP C
BA	CP D
BB	CP E
BC	CP H

オペコード	ニーモニックコード
BD	CP L
BE	CP (HL)
BF	CP A
C0	RET NZ
C1	POP BC
C2*8405*	JP NZ,nn
C3*8405*	JP nn
C4*8405*	CALL NZ,nn
C5	PUSH BC
C6*20*	ADD A,n
C7	RST 00H
C8	RET Z
C9	RET
CA*8405*	JP Z,nn
CC*8405*	CALL Z,nn
CD*8405*	CALL nn
CE*20*	ADC A,n
CF	RST 08H
D0	RET NC
D1	POP DE
D2*8405*	JP NC,nn
D3*20*	OUT (n),A
D4*8405*	CALL NC,nn
D5	PUSH DE
D6*20*	SUB n
D7	RST 10H
D8	RET C
D9	EXX
DA*8405*	JP C,nn
DB*20*	IN A,(n)
DC*8405*	CALL C,nn
DE*20*	SBC A,n
DF	RST 18H
E0	RET PO
E1	POP HL
E2*8405*	JP PO,nn
E3	EX (SP),HL

オペコード	ニーモニックコード
E4*8405*	CALL PO,nn
E5	PUSH HL
E6*20*	AND n
E7	RST 20H
E8	RET PE
E9	JP (HL)
EA*8405*	JP PE,nn
EB	EX DE,HL
EC*8405*	CALL PE,nn
EE*20*	XOR n
EF	RST 28H
F0	RET P
F1	POP AF
F2*8405*	JP P,nn
F3	DI
F4*8405*	CALL P,nn
F5	PUSH AF
F6*20*	OR n
F7	RST 30H
F8	RET M
F9	LD SP,HL
FA*8405*	JP M,nn
FB	EI
FC*8405*	CALL M,nn
FE*20*	CP n
FF	RST 38H
CB00	RLC B
CB01	RLC C
CB02	RLC D
CB03	RLC E
CB04	RLC H
CB05	RLC L
CB06	RLC (HL)
CB07	RLC A
CB08	RRC B
CB09	RRC C
CB0A	RRC D
CB0B	RRC E

オペコード	ニーモニックコード
CB0C	RRC H
CB0D	RRC L
CB0E	RRC (HL)
CB0F	RRC A
CB10	RL B
CB11	RL C
CB12	RL D
CB13	RL E
CB14	RL H
CB15	RL L
CB16	RL (HL)
CB17	RL A
CB18	RR B
CB19	RR C
CB1A	RR D
CB1B	RR E
CB1C	RR H
CB1D	RR L
CB1E	RR (HL)
CB1F	RR A
CB20	SLA B
CB21	SLA C
CB22	SLA D
CB23	SLA E
CB24	SLA H
CB25	SLA L
CB26	SLA (HL)
CB27	SLA A
CB28	SRA B
CB29	SRA C
CB2A	SRA D
CB2B	SRA E
CB2C	SRA H
CB2D	SRA L
CB2E	SRA (HL)
CB2F	SRA A
CB38	SRL B

オペコード	ニーモニックコード
CB39	SRL C
CB3A	SRL D
CB3B	SRL E
CB3C	SRL H
CB3D	SRL L
CB3E	SRL (HL)
CB3F	SRL A
CB40	BIT 0,B
CB41	BIT 0,C
CB42	BIT 0,D
CB43	BIT 0,E
CB44	BIT 0,H
CB45	BIT 0,L
CB46	BIT 0,(HL)
CB47	BIT 0,A
CB48	BIT 1,B
CB49	BIT 1,C
CB4A	BIT 1,D
CB4B	BIT 1,E
CB4C	BIT 1,H
CB4D	BIT 1,L
CB4E	BIT 1,(HL)
CB4F	BIT 1,A
CB50	BIT 2,B
CB51	BIT 2,C
CB52	BIT 2,D
CB53	BIT 2,E
CB54	BIT 2,H
CB55	BIT 2,L
CB56	BIT 2,(HL)
CB57	BIT 2,A
CB58	BIT 3,B
CB59	BIT 3,C
CB5A	BIT 3,D
CB5B	BIT 3,E
CB5C	BIT 3,H
CB5D	BIT 3,L
CB5E	BIT 3,(HL)

オペコード	ニーモニックコード
CB5F	BIT 3,A
CB60	BIT 4,B
CB61	BIT 4,C
CB62	BIT 4,D
CB63	BIT 4,E
CB64	BIT 4,H
CB65	BIT 4,L
CB66	BIT 4,(HL)
CB67	BIT 4,A
CB68	BIT 5,B
CB69	BIT 5,C
CB6A	BIT 5,D
CB6B	BIT 5,E
CB6C	BIT 5,H
CB6D	BIT 5,L
CB6E	BIT 5,(HL)
CB6F	BIT 5,A
CB70	BIT 6,B
CB71	BIT 6,C
CB72	BIT 6,D
CB73	BIT 6,E
CB74	BIT 6,H
CB75	BIT 6,L
CB76	BIT 6,(HL)
CB77	BIT 6,A
CB78	BIT 7,B
CB79	BIT 7,C
CB7A	BIT 7,D
CB7B	BIT 7,E
CB7C	BIT 7,H
CB7D	BIT 7,L
CB7E	BIT 7,(HL)
CB7F	BIT 7,A
CB80	RES 0,B
CB81	RES 0,C
CB82	RES 0,D
CB83	RES 0,E

オペコード	ニーモニックコード
CB84	RES 0,H
CB85	RES 0,L
CB86	RES 0,(HL)
CB87	RES 0,A
CB88	RES 1,B
CB89	RES 1,C
CB8A	RES 1,D
CB8B	RES 1,E
CB8C	RES 1,H
CB8D	RES 1,L
CB8E	RES 1,(HL)
CB8F	RES 1,A
CB90	RES 2,B
CB91	RES 2,C
CB92	RES 2,D
CB93	RES 2,E
CB94	RES 2,H
CB95	RES 2,L
CB96	RES 2,(HL)
CB97	RES 2,A
CB98	RES 3,B
CB99	RES 3,C
CB9A	RES 3,D
CB9B	RES 3,E
CB9C	RES 3,H
CB9D	RES 3,L
CB9E	RES 3,(HL)
CB9F	RES 3,A
CBA0	RES 4,B
CBA1	RES 4,C
CBA2	RES 4,D
CBA3	RES 4,E
CBA4	RES 4,H
CBA5	RES 4,L
CBA6	RES 4,(HL)
CBA7	RES 4,A
CBA8	RES 5,B
CBA9	RES 5,C

オペコード	ニーモニックコード
CBAA	RES 5,D
CBAB	RES 5,E
CBAC	RES 5,H
CBAD	RES 5,L
CBAE	RES 5,(HL)
CBAF	RES 5,A
CBB0	RES 6,B
CBB1	RES 6,C
CBB2	RES 6,D
CBB3	RES 6,E
CBB4	RES 6,H
CBB5	RES 6,L
CBB6	RES 6,(HL)
CBB7	RES 6,A
CBB8	RES 7,B
CBB9	RES 7,C
CBBA	RES 7,D
CBBB	RES 7,E
CBBC	RES 7,H
CBBD	RES 7,L
CBBE	RES 7,(HL)
CBBF	RES 7,A
CBC0	SET 0,B
CBC1	SET 0,C
CBC2	SET 0,D
CBC3	SET 0,E
CBC4	SET 0,H
CBC5	SET 0,L
CBC6	SET 0,(HL)
CBC7	SET 0,A
CBC8	SET 1,B
CBC9	SET 1,C
CBCA	SET 1,D
CBCB	SET 1,E
CBCC	SET 1,H
CBCD	SET 1,L
CBCE	SET 1,(HL)
CBCF	SET 1,A

オペコード	ニーモニックコード
CBD0	SET 2,B
CBD1	SET 2,C
CBD2	SET 2,D
CBD3	SET 2,E
CBD4	SET 2,H
CBD5	SET 2,L
CBD6	SET 2,(HL)
CBD7	SET 2,A
CBD8	SET 3,B
CBD9	SET 3,C
CBDA	SET 3,D
CBDB	SET 3,E
CBDC	SET 3,H
CBDD	SET 3,L
CBDE	SET 3,(HL)
CBDF	SET 3,A
CBE0	SET 4,B
CBE1	SET 4,C
CBE2	SET 4,D
CBE3	SET 4,E
CBE4	SET 4,H
CBE5	SET 4,L
CBE6	SET 4,(HL)
CBE7	SET 4,A
CBE8	SET 5,B
CBE9	SET 5,C
CBEA	SET 5,D
CBEB	SET 5,E
CBEC	SET 5,H
CBED	SET 5,L
CBEE	SET 5,(HL)
CBEF	SET 5,A
CBF0	SET 6,B
CBF1	SET 6,C
CBF2	SET 6,D
CBF3	SET 6,E
CBF4	SET 6,H
CBF5	SET 6,L

オペコード	ニーモニックコード
CBF6	SET 6,(HL)
CBF7	SET 6,A
CBF8	SET 7,B
CBF9	SET 7,C
CBFA	SET 7,D
CBFB	SET 7,E
CBFC	SET 7,H
CBFD	SET 7,L
CBFE	SET 7,(HL)
CBFF	SET 7,A
DD09	ADD IX,BC
DD19	ADD IX,DE
DD21 *8405*	LD IX,nn
DD22 *8405*	LD (nn),IX
DD23	INC IX
DD29	ADD IX,IX
DD2A*8405*	LD IX,(nn)
DD2B	DEC IX
DD34 *05*	INC (IX+d)
DD35 *05*	DEC (IX+d)
DD36 *0520*	LD (IX+d),n
DD39	ADD IX,SP
DD46 *05*	LD B,(IX+d)
DD4E*05*	LD C,(IX+d)
DD56 *05*	LD D,(IX+d)
DD5E*05*	LD E,(IX+d)
DD66 *05*	LD H,(IX+d)
DD6E*05*	LD L,(IX+d)
DD70 *05*	LD (IX+d),B
DD71 *05*	LD (IX+d),C
DD72 *05*	LD (IX+d),D
DD73 *05*	LD (IX+d),E
DD74 *05*	LD (IX+d),H
DD75 *05*	LD (IX+d),L
DD77 *05*	LD (IX+d),A
DD7E*05*	LD A,(IX+d)
DD86 *05*	ADD A,(IX+d)
DD8E*05*	ADC A,(IX+d)
DD96 *05*	SUB (IX+d)

オペコード	ニーモニックコード
DD9E *05*	SBC A,(IX+d)
DDA6 *05*	AND (IX+d)
DDAE*05*	XOR (IX+d)
DDB6 *05*	OR (IX+d)
DDBE*05*	CP (IX+d)
DDE1	POP IX
DDE3	EX (SP),IX
DDE5	PUSH IX
DDE9	JP (IX)
DDF9	LD SP,IX
DDCB*05*06	RLC (IX+d)
DDCB*05*0E	RRC (IX+d)
DDCB*05*16	RL (IX+d)
DDCB*05*1E	RR (IX+d)
DDCB*05*26	SLA (IX+d)
DDCB*05*2E	SRA (IX+d)
DDCB*05*3E	SRL (IX+d)
DDCB*05*46	BIT 0,(IX+d)
DDCB*05*4E	BIT 1,(IX+d)
DDCB*05*56	BIT 2,(IX+d)
DDCB*05*5E	BIT 3,(IX+d)
DDCB*05*66	BIT 4,(IX+d)
DDCB*05*6E	BIT 5,(IX+d)
DDCB*05*76	BIT 6,(IX+d)
DDCB*05*7E	BIT 7,(IX+d)
DDCB*05*86	RES 0,(IX+d)
DDCB*05*8E	RES 1,(IX+d)
DDCB*05*96	RES 2,(IX+d)
DDCB*05*9E	RES 3,(IX+d)
DDCB*05*A6	RES 4,(IX+d)
DDCB*05*AE	RES 5,(IX+d)
DDCB*05*B6	RES 6,(IX+d)
DDCB*05*BE	RES 7,(IX+d)
DDCB*05*C6	SET 0,(IX+d)
DDCB*05*CE	SET 1,(IX+d)
DDCB*05*D6	SET 2,(IX+d)
DDCB*05*DE	SET 3,(IX+d)
DDCB*05*E6	SET 4,(IX+d)
DDCB*05*EE	SET 5,(IX+d)

オペコード	ニーモニックコード
DDCB05F6	SET 6,(IX+d)
DDCB05FE	SET 7,(IX+d)
ED40	IN B,(C)
ED41	OUT (C),B
ED42	SBC HL,BC
ED438405	LD (nn),BC
ED44	NEG
ED45	RETN
ED46	IM 0
ED47	LD I,A
ED48	IN C,(C)
ED49	OUT (C),C
ED4A	ADC HL,BC
ED4B8405	LD BC,(nn)
ED4D	RETI
ED50	IN D,(C)
ED51	OUT (C),D
ED52	SBC HL,DE
ED538405	LD (nn),DE
ED56	IM 1
ED57	LD A,I
ED58	IN E,(C)
ED59	OUT (C),E
ED5A	ADC HL,DE
ED5B8405	LD DE,(nn)
ED5E	IM 2
ED60	IN H,(C)
ED61	OUT (C),H
ED62	SBC HL,HL
ED67	RRD
ED68	IN L,(C)
ED69	OUT (C),L
ED6A	ADC HL,HL
ED6F	RLD
ED72	SBC HL,SP
ED738405	LD (nn),SP
ED78	IN A,(C)
ED79	OUT (C),A
ED7A	ADC HL,SP

オペコード	ニーモニックコード
ED7B8405	LD SP,(nn)
EDA0	LDI
EDA1	CPI
EDA2	INI
EDA3	OUTI
EDA8	LDD
EDA9	CPD
EDAA	IND
EDAB	OUTD
EDB0	LDIR
EDB1	CPIR
EDB2	INIR
EDB3	OTIR
EDB8	LDDR
EDB9	CPDR
EDBA	INDR
EDBB	OTDR
FD09	ADD IY,BC
FD19	ADD IY,DE
FD218405	LD IY,nn
FD228405	LD (nn),IY
FD23	INC IY
FD29	ADD IY,IY
FD2A8405	LD IY,(nn)
FD2B	DEC IY
FD3405	INC (IY+d)
FD3505	DEC (IY+d)
FD360520	LD (IY+d),n
FD39	ADD IY,SP
FD4605	LD B,(IY+d)
FD4E05	LD C,(IY+d)
FD5605	LD D,(IY+d)
FD5E05	LD E,(IY+d)
FD6605	LD H,(IY+d)
FD6E05	LD L,(IY+d)
FD7005	LD (IY+d),B
FD7105	LD (IY+d),C
FD7205	LD (IY+d),D
FD7305	LD (IY+d),E

オペコード	ニーモニックコード
FD74 *05*	LD (IY+d),H
FD75 *05*	LD (IY+d),L
FD77 *05*	LD (IY+d),A
FD7E *05*	LD A,(IY+d)
FD86 *05*	ADD A,(IY+d)
FD8E *05*	ADC A,(IY+d)
FD96 *05*	SUB (IY+d)
FD9E *05*	SBC A,(IY+d)
FDA6 *05*	AND (IY+d)
FDAE*05*	XOR (IY+d)
FDB6 *05*	OR (IY+d)
FDBE*05*	CP (IY+d)
FDE1	POP IY
FDE3	EX (SP),IY
FDE5	PUSH IY
FDE9	JP (IY)
FDF9	LD SP,IY
FDCB*05*06	RLC (IY+d)
FDCB*05*0E	RRC (IY+d)
FDCB*05*16	RL (IY+d)
FDCB*05*1E	RR (IY+d)
FDCB*05*26	SLA (IY+d)
FDCB*05*2E	SRA (IY+d)
FDCB*05*3E	SRL (IY+d)
FDCB*05*46	BIT 0,(IY+d)
FDCB*05*4E	BIT 1,(IY+d)
FDCB*05*56	BIT 2,(IY+d)
FDCB*05*5E	BIT 3,(IY+d)
FDCB*05*66	BIT 4,(IY+d)
FDCB*05*6E	BIT 5,(IY+d)
FDCB*05*76	BIT 6,(IY+d)
FDCB*05*7E	BIT 7,(IY+d)
FDCB*05*86	RES 0,(IY+d)
FDCB*05*8E	RES 1,(IY+d)
FDCB*05*96	RES 2,(IY+d)
FDCB*05*9E	RES 3,(IY+d)
FDCB*05*A6	RES 4,(IY+d)
FDCB*05*AE	RES 5,(IY+d)
FDCB*05*B6	RES 6,(IY+d)

オペコード	ニーモニックコード
FDCB*05*BE	RES 7,(IY+d)
FDCB*05*C6	SET 0,(IY+d)
FDCB*05*CE	SET 1,(IY+d)
FDCB*05*D6	SET 2,(IY+d)
FDCB*05*DE	SET 3,(IY+d)
FDCB*05*E6	SET 4,(IY+d)
FDCB*05*EE	SET 5,(IY+d)
FDCB*05*F6	SET 6,(IY+d)
FDCB*05*FE	SET 7,(IY+d)

A.3 IPLプログラム・アセンブリ・リスト

MZ-2000のIPL (Initial Program Loader) の全プログラムのアセンブリ・リストを掲載しています。アセンブリ・リストはZ80A-CPUアセンブラ (MZ-80シリーズFDOS中のシステムプログラム) によるプリンタ出力リストです。

IPLの動作と一般の操作方法については、Owner's Manual第2章に述べられています。本リストは、IPLの動作の詳細を調べるための参考データであり、IPLプログラムのアセンブリ・リストの内容についてのお問い合わせには応じかねます。

IPLのブートローディング機能は、3種類の外部ファイルを対象としています。第1は、MZ-2000本体に装備されているカセットテープデッキによって、カセットテープファイル、たとえば付属のBASICインタープリタを起動する働き、第2にオプション周辺機器であるミニフロッピーディスクドライブによって、ディスケットファイル、たとえばDISK BASICインタープリタ、FDOS等を起動する働き、第3に、I/Oポート上にシステムプログラムのROMカード等を置いて、そのシステムプログラムを起動するための機能とが備えられています。

上記のそれぞれのルーチンは、アセンブリ・リスト上のリマーク表示で、

CMT CONTROL (P.150)
MFM MINIFLOPPY CONTROL (P.156)
INTRAM-EXROM (P.162)

によってそれぞれ場所が示されています。

```
0000                ;************************
0000                ;
0000                ;    Personal Computer
0000                ;        MZ-2000
0000                ;
0000                ;        Initial
0000                ;        Program
0000                ;        Loader
0000                ;
0000                ;************************
0000                ;
0000 1804                   JR      START
0002                ;;;;;;;;;;;;;;;
0002                ; NST RESET
0002                ;
0002 3E03           NST:    LD      A,3H
0004 D3E3                   OUT     (E3H),A
0006                ;;;;;;;;;;;;;;;
0006                ;INITIALIZE
0006                ;
0006 3E82           START:  LD      A,82H           ;8255 A=OUT B=IN C=OUT
0008 D3E3                   OUT     (E3H),A
000A 3E58                   LD      A,58H           ;BST=1 OPEN=1 WRITE=1
000C D3E2                   OUT     (E2H),A
000E 31E0FF                 LD      SP,FFE0H
0011 3EF7                   LD      A,F7H           ;STOP
0013 D3E0                   OUT     (E0H),A
0015 3E0F                   LD      A,0FH           ;PIO A=OUT
0017 D3E9                   OUT     (E9H),A
0019 3ECF                   LD      A,CFH           ;PIO B=IN
001B D3EB                   OUT     (EBH),A
001D 3EFF                   LD      A,FFH
001F D3EB                   OUT     (EBH),A
0021 AF                     XOR     A
0022 D3F6                   OUT     (F6H),A
0024 D3F4                   OUT     (F4H),A
0026 3C                     INC     A
0027 D3F7                   OUT     (F7H),A
0029 3E07                   LD      A,07H
002B D3F5                   OUT     (F5H),A
002D 3ED3                   LD      A,D3H
002F D3E8                   OUT     (E8H),A
0031 2100D0                 LD      HL,D000H
0034 3ED8                   LD      A,D8H
0036 3600           CLEAR:  LD      (HL),00H        ;DISPLAY CLEAR
0038 23                     INC     HL
0039 BC                     CP      H
003A 20FA                   JR      NZ,CLEAR
003C 3EFF                   LD      A,FFH           ;ALL STOP
003E D3E0                   OUT     (E0H),A
0040 3E03                   LD      A,03H
0042 D3F7                   OUT     (F7H),A
0044 CD6E00                 CALL    GCLEAR
0047 3E02                   LD      A,02H
0049 D3F7                   OUT     (F7H),A
004B CD6E00                 CALL    GCLEAR
004E 3E01                   LD      A,01H
0050 D3F7                   OUT     (F7H),A
0052 CD6E00                 CALL    GCLEAR
0055 3E13                   LD      A,13H
0057 D3E8                   OUT     (E8H),A
0059 AF                     XOR     A
005A 32ECFF                 LD      (DRINO),A
005D 32E6FF                 LD      (MTFG),A
0060 CD8C00         KEYIN:  CALL    KEYS1
0063 CB5F                   BIT     3,A
0065 2845                   JR      Z,CMT
0067 CB47                   BIT     0,A
0069 CA1706                 JP      Z,EXROMT
006C 182A                   JR      NKIN
```

```
006E                  ;
006E 2100C0           GCLEAR: LD      HL,C000H        ;G-RAM CLEAR
0071 F3                       DI
0072 DBE8                     IN      A,(E8H)
0074 CBFF                     SET     7,A
0076 CBB7                     RES     6,A
0078 D3E8                     OUT     (E8H),A
007A 1101C0                   LD      DE,C001H
007D 3600                     LD      (HL),00H
007F 017F3E                   LD      BC,3E7FH
0082 EDB0                     LDIR
0084 CBBF                     RES     7,A
0086 CBF7                     SET     6,A
0088 D3E8                     OUT     (E8H),A
008A FB                       EI
008B C9                       RET
008C                  ;
008C 0614             KEYS1:  LD      B,14H           ;KEY STROBE OUT
008E DBE8             KEYS:   IN      A,(E8H)
0090 E6F0                     AND     F0H
0092 B0                       OR      B
0093 D3E8                     OUT     (E8H),A
0095 DBEA                     IN      A,(EAH)
0097 C9                       RET
0098                  ;
0098                  ;
0098 CDA000           NKIN:   CALL    FDCC
009B CA6403                   JP      Z,FD
009E 180C                     JR      CMT
00A0                  ;
00A0 3EA5             FDCC:   LD      A,A5H
00A2 47                       LD      B,A
00A3 D3D9                     OUT     (D9H),A
00A5 CDFE05                   CALL    DLY80U
00A8 DBD9                     IN      A,(D9H)
00AA B8                       CP      B
00AB C9                       RET
00AC                  ;;;;;;;;;;;;;;;;
00AC                  ;                 ;
00AC                  ;   CMT CONTROL   ;
00AC                  ;                 ;
00AC                  ;;;;;;;;;;;;;;;;
00AC CDF601           CMT:    CALL    MSTOP
00AF CD0E02                   CALL    KEYMES
00B2 CDE900                   CALL    ?RDI
00B5 3817                     JR      C,ST1
00B7 CD5702                   CALL    LDMSG
00BA 2101CF                   LD      HL,NAME
00BD 1E10                     LD      E,10H
00BF 0E10                     LD      C,10H
00C1 CD6002                   CALL    DISP2
00C4 3A00CF                   LD      A,(ATRB)
00C7 FE01                     CP      1
00C9 200E                     JR      NZ,MISMCH
00CB CD1001                   CALL    ?RDD
00CE F5               ST1:    PUSH    AF
00CF CD2502                   CALL    REW
00D2 F1                       POP     AF
00D3 DA8705                   JP      C,TRYAG
00D6 C30200                   JP      NST
00D9                  ;
00D9 214E03           MISMCH: LD      HL,MES16
00DC 1E0A                     LD      E,0AH
00DE 0E0F                     LD      C,15
00E0 CD6D02                   CALL    DISPCS
00E3 CDF601                   CALL    MSTOP
00E6 37                       SCF
00E7 18E5                     JR      ST1
00E9                  ;
00E9                  ;READ INFORMATION
```

```
00E9                    ;       CF=1:ERROR
00E9                    RDINF:  ENT
00E9 F3                 ?RDI:   DI
00EA DBE2                       IN      A,(E2H)
00EC CBEF                       SET     5,A
00EE D3E2                       OUT     (E2H),A
00F0 1604                       LD      D,4
00F2 018000                     LD      BC,0080H
00F5 2100CF                     LD      HL,IBUFE
00F8 CDC701             RD1:    CALL    MOTOR
00FB 380E                       JR      C,STPEIR
00FD CD9301                     CALL    TMARK
0100 3809                       JR      C,STPEIR
0102 CD1C01                     CALL    RTAPE
0105 3804                       JR      C,STPEIR
0107 CB5A               RET2S:  BIT     3,D
0109 2803                       JR      Z,EIRTN
010B CDF601             STPEIR: CALL    MSTOP
010E FB                 EIRTN:  EI
010F C9                         RET
0110                    ;
0110                    ;
0110                    ;READ DATA
0110                    RDDAT:  ENT
0110 F3                 ?RDD:   DI
0111 1608                       LD      D,8
0113 ED4B12CF                   LD      BC,(SIZE)
0117 210080                     LD      HL,8000H
011A 18DC                       JR      RD1
011C                    ;
011C                    ;
011C                    ;READ TAPE
011C                    ;       BC=SIZE
011C                    ;       DE=LOAD ADDRESS
011C D5                 RTAPE:  PUSH    DE
011D C5                         PUSH    BC
011E E5                         PUSH    HL
011F 2602                       LD      H,2
0121 CDBB01             RTP2:   CALL    SPDIN
0124 3838                       JR      C,TRTN1
0126 28F9                       JR      Z,RTP2
0128 54                         LD      D,H
0129 210000                     LD      HL,0000H
012C 22E0FF                     LD      (SUMDT),HL
012F E1                         POP     HL
0130 C1                         POP     BC
0131 C5                         PUSH    BC
0132 E5                         PUSH    HL
0133 CD7301             RTP3:   CALL    RBYTE
0136 3826                       JR      C,TRTN1
0138 77                         LD      (HL),A
0139 23                         INC     HL
013A 0B                         DEC     BC
013B 78                         LD      A,B
013C B1                         OR      C
013D 20F4                       JR      NZ,RTP3
013F 2AE0FF                     LD      HL,(SUMDT)
0142 CD7301                     CALL    RBYTE
0145 3817                       JR      C,TRTN1
0147 5F                         LD      E,A
0148 CD7301                     CALL    RBYTE
014B 3811                       JR      C,TRTN1
014D BD                         CP      L
014E 2004                       JR      NZ,RTP5
0150 7B                         LD      A,E
0151 BC                         CP      H
0152 280A                       JR      Z,TRTN1
0154 15                 RTP5:   DEC     D
0155 2803                       JR      Z,RTP6
0157 62                         LD      H,D
```

```
0158 18C7               JR     RTP2
015A CD6602     RTP6:   CALL   BOOTER
015D 37                 SCF
015E E1         TRTN1:  POP    HL
015F C1                 POP    BC
0160 D1                 POP    DE
0161 C9                 RET
0162            ;EDGE
0162 DBE1       EDGE:   IN     A,(E1H)
0164 2F                 CPL
0165 07                 RLCA
0166 D8                 RET    C
0167 07                 RLCA
0168 30F8               JR     NC,EDGE
016A DBE1       EDGE1:  IN     A,(E1H)
016C 2F                 CPL
016D 07                 RLCA
016E D8                 RET    C
016F 07                 RLCA
0170 38F8               JR     C,EDGE1
0172 C9                 RET
0173            ; 1 BYTE READ
0173            ;       DATA=A
0173            ;       SUMDT STORE
0173 E5         RBYTE:  PUSH   HL
0174 210008             LD     HL,0800H
0177 CDBB01     RBY1:   CALL   SPDIN
017A 3815               JR     C,RBY3
017C 280A               JR     Z,RBY2
017E E5                 PUSH   HL
017F 2AE0FF             LD     HL,(SUMDT)
0182 23                 INC    HL
0183 22E0FF             LD     (SUMDT),HL
0186 E1                 POP    HL
0187 37                 SCF
0188 CB15       RBY2:   RL     L
018A 25                 DEC    H
018B 20EA               JR     NZ,RBY1
018D CD6201             CALL   EDGE
0190 7D                 LD     A,L
0191 E1         RBY3:   POP    HL
0192 C9                 RET
0193            ;TAPE MARK DETECT
0193            ;       E=L:INFORMATION
0193            ;       E=S:DATA
0193 E5         TMARK:  PUSH   HL
0194 211414             LD     HL,1414H
0197 CB5A               BIT    3,D
0199 2001               JR     NZ,TM0
019B 29                 ADD    HL,HL
019C 22E2FF     TM0:    LD     (TMCNT),HL
019F 2AE2FF     TM1:    LD     HL,(TMCNT)
01A2 CDBB01     TM2:    CALL   SPDIN
01A5 38EA               JR     C,RBY3
01A7 28F6               JR     Z,TM1
01A9 25                 DEC    H
01AA 20F6               JR     NZ,TM2
01AC CDBB01     TM3:    CALL   SPDIN
01AF 38E0               JR     C,RBY3
01B1 20EC               JR     NZ,TM1
01B3 2D                 DEC    L
01B4 20F6               JR     NZ,TM3
01B6 CD6201             CALL   EDGE
01B9 18D6               JR     RBY3
01BB            ;
01BB CD6201     SPDIN:  CALL   EDGE
01BE D8                 RET    C
01BF CD5002             CALL   DLY2
01C2 DBE1               IN     A,(E1H)
01C4 E640               AND    40H
```

```
01C6 C9                         RET
01C7                    ;
01C7                    ;
01C7                    ;MOTOR ON
01C7 D5                 MOTOR:  PUSH    DE
01C8 C5                         PUSH    BC
01C9 E5                         PUSH    HL
01CA DBE1                       IN      A,(E1H)
01CC E620                       AND     20H
01CE 281F                       JR      Z,MOTRD
01D0 21B302                     LD      HL,MES6
01D3 1E0A                       LD      E,AH
01D5 0E0E                       LD      C,14D
01D7 CD6D02                     CALL    DISPCS
01DA CD0202                     CALL    OPEN
01DD DBEA               MOT1:   IN      A,(EAH)
01DF 2F                         CPL
01E0 07                         RLCA
01E1 380F                       JR      C,MOTR
01E3 DBE1                       IN      A,(E1H)
01E5 E620                       AND     20H
01E7 20F4                       JR      NZ,MOT1
01E9 CD0E02                     CALL    KEYMES
01EC CD4A02                     CALL    DEL1M
01EF CD1902             MOTRD:  CALL    PLAY
01F2 E1                 MOTR:   POP     HL
01F3 C1                         POP     BC
01F4 D1                         POP     DE
01F5 C9                         RET
01F6                    ;
01F6                    ;
01F6                    ;MOTOR STOP
01F6 3EF7               MSTOP:  LD      A,F7H
01F8 D3E0                       OUT     (E0H),A
01FA CD4402                     CALL    DEL6
01FD 3EFF                       LD      A,FFH
01FF D3E0                       OUT     (E0H),A
0201 C9                         RET
0202                    ;EJECT
0202 3E08               OPEN:   LD      A,08H
0204 D3E3                       OUT     (E3H),A
0206 CD4402                     CALL    DEL6
0209 3E09                       LD      A,09H
020B D3E3                       OUT     (E3H),A
020D C9                         RET
020E                    ;
020E                    ;
020E 219702             KEYMES: LD      HL,MES3
0211 1E04                       LD      E,4H
0213 0E1C                       LD      C,28D
0215 CD6D02                     CALL    DISPCS
0218 C9                         RET
0219                    ;
0219                    ;PLAY
0219 3EFB               PLAY:   LD      A,FBH
021B D3E0                       OUT     (E0H),A
021D CD4402                     CALL    DEL6
0220 3EFF                       LD      A,FFH
0222 D3E0                       OUT     (E0H),A
0224 C9                         RET
0225                    ;
0225                    ;
0225                    ;REWIND
0225 3EFE               REW:    LD      A,FEH
0227 D3E0                       OUT     (E0H),A
0229 CD4402                     CALL    DEL6
022C 3EFF                       LD      A,FFH
022E D3E0                       OUT     (E0H),A
0230 DBE2                       IN      A,(E2H)
0232 CBAF                       RES     5,A
```

```
0234 D3E2                OUT     (E2H),A
0236 C9                  RET
0237                ;
0237                ;TIMING DEL
0237 F5             D1M:    PUSH    AF
0238 AF                  XOR     A
0239 3D                  DEC     A
023A 20FD                JR      NZ,-1
023C 0B                  DEC     BC
023D 78                  LD      A,B
023E B1                  OR      C
023F 20F7                JR      NZ,D1M+1
0241 F1                  POP     AF
0242 C1                  POP     BC
0243 C9                  RET
0244 C5             DEL6:   PUSH    BC
0245 01E900              LD      BC,233
0248 18ED                JR      D1M
024A C5             DEL1M:  PUSH    BC
024B 010F06              LD      BC,1551
024E 18E7                JR      D1M
0250                ;
0250                ;TAPE DELY TIMING
0250                ;
0250                ;
0250 3E31           DLY2:   LD      A,31H
0252 3D                  DEC     A
0253 C25202              JP      NZ,DLY2+2
0256 C9                  RET
0257                ;
0257                ;
0257                ;
0257                ;;;;;;;;;;;;;;;;;;;;;;;;;;
0257  P             IBUFE:  EQU     CF00H
0257  P             ATRB:   EQU     CF00H
0257  P             NAME:   EQU     CF01H
0257  P             SIZE:   EQU     CF12H
0257  P             DTADR:  EQU     CF14H
0257  P             SUMDT:  EQU     FFE0H
0257  P             TMCNT:  EQU     FFE2H
0257                ;;;;;;;;;;;;;;;;;;;;;;;;;;
0257                ;
0257 218902         LDMSG:  LD      HL,MES1
025A 1E00                LD      E,0H
025C 0E0E                LD      C,14D
025E 180D                JR      DISPCS
0260                ;
0260 3ED3           DISP2:  LD      A,D3H
0262 D3E8                OUT     (E8H),A
0264 1818                JR      DISP1
0266                ;
0266 21C102         BOOTER: LD      HL,MES8
0269 1E0A                LD      E,AH
026B 0E0D                LD      C,13D
026D                ;
026D 3ED3           DISPCS: LD      A,D3H
026F D3E8                OUT     (E8H),A
0271 D9                  EXX
0272 2100D0              LD      HL,D000H
0275 3ED8                LD      A,D8H
0277 3600           DISP3:  LD      (HL),00H
0279 23                  INC     HL
027A BC                  CP      H
027B 20FA                JR      NZ,DISP3
027D D9                  EXX
027E AF             DISP1:  XOR     A
027F 47                  LD      B,A
0280 16D0                LD      D,D0H
0282 EDB0                LDIR
0284 3E13                LD      A,13H
```

```
0286 D3E8                   OUT     (E8H),A
0288 C9                     RET
0289                ;
0289                ;
0289 49504C20       MES1:   DEFM    'IPL is loading'
028D 6973206C
0291 6F616469
0295 6E67
0297 49504C20       MES3:   DEFM    'IPL is looking for a program'
029B 6973206C
029F 6F6F6B69
02A3 6E672066
02A7 6F722061
02AB 2070726F
02AF 6772616D
02B3 4D616B65       MES6:   DEFM    'Make ready CMT'
02B7 20726561
02BB 64792043
02BF 4D54
02C1 4C6F6164       MES8:   DEFM    'Loading error'
02C5 696E6720
02C9 6572726F
02CD 72
02CE 4D616B65       MES9:   DEFM    'Make ready FD'
02D2 20726561
02D6 64792046
02DA 44
02DB 50726573       MES10:  DEFM    'Press F or C'
02DF 73204620
02E3 6F722043
02E7 463A466C       MES11:  DEFM    'F:Floppy diskette'
02EB 6F707079
02EF 20646973
02F3 6B657474
02F7 65
02F8 433A4361       MES12:  DEFM    'C:Cassette tape'
02FC 73736574
0300 74652074
0304 617065
0307 44726976       MES13:  DEFM    'Drive No? (1-4)'
030B 65204E6F
030F 3F202831
0313 2D3429
0316 54686973       MES14:  DEFM    'This diskette is not master'
031A 20646973
031E 6B657474
0322 65206973
0326 206E6F74
032A 206D6173
032E 746572
0331 50726573       MES15:  DEFM    'Pressing S key starts the CMT'
0335 73696E67
0339 2053206B
033D 65792073
0341 74617274
0345 73207468
0349 6520434D
034D 54
034E 46696C65       MES16:  DEFM    'File mode error'
0352 206D6F64
0356 65206572
035A 726F72
035D                ;
035D 01             IPLMC:  DEFB    01H
035E 49504C50               DEFM    'IPLPRO'
0362 524F
0364                ;
0364                ;
0364                ;
0364                ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
0364                    ;                          ;
0364                    ;  MFM MINIFLOPPY CONTROL  ;
0364                    ;                          ;
0364                    ;;;;;;;;;;;;;;;;;;;;;;;;;;;;
0364                    ;
0364                    ;  CASE OF DISK INITIALIZE
0364                    ;     DRIVE NO=DRINO (0-3)
0364                    ;
0364                    ;  CASE OF SEQUENTIAL READ
0364                    ;     DRIVE NO=DRINO (0-3)
0364                    ;     BYTE SIZE     =IY+2,3
0364                    ;     ADDRESS       =IX+0,1
0364                    ;     NEXT TRACK    =IY+0
0364                    ;     NEXT SECTOR   =IY+1
0364                    ;     START TRACK   =IY+4
0364                    ;     START SECTOR  =IY+5
0364                    ;
0364                    ;
0364                    ; I/O PORT ADDRESS
0364                    ;
0364  P                 CR:     EQU    D8H          ;STATUS/COMAND PORT
0364  P                 TR:     EQU    D9H          ;TRACK REG PORT
0364  P                 SCR:    EQU    DAH          ;SECTOR REG PORT
0364  P                 DR:     EQU    DBH          ;DATA REG PORT
0364  P                 DM:     EQU    DCH          ;MOTOR/DRIVE PORT
0364  P                 HS:     EQU    DDH          ;HEAD SIDE SELECT PORT
0364                    ;
0364                    ;;;;;;;;;;
0364                    ;FD
0364 DD2100CF           FD:     LD     IX,IBADR1
0368 AF                         XOR    A
0369 321ECF                     LD     (CF1EH),A
036C 321FCF                     LD     (CF1FH),A
036F FD21E0FF                   LD     IY,FFE0H
0373 210001                     LD     HL,0100H
0376 FD7502                     LD     (IY+2),L
0379 FD7403                     LD     (IY+3),H
037C CDA204                     CALL   BREAD        ;INFORMATION INPUT
037F 2100CF                     LD     HL,CF00H     ;MASTER CHECK
0382 115D03                     LD     DE,IPLMC
0385 0606                       LD     B,6
0387 4E                 MCHECK: LD     C,(HL)
0388 1A                         LD     A,(DE)
0389 B9                         CP     C
038A C27205                     JP     NZ,NMASTE
038D 23                         INC    HL
038E 13                         INC    DE
038F 10F6                       DJNZ   MCHECK
0391 CD5702                     CALL   LDMSG
0394 2107CF                     LD     HL,CF07H
0397 1E10                       LD     E,10H
0399 0E0A                       LD     C,AH
039B CD6002                     CALL   DISP2
039E DD210080                   LD     IX,IBADR2
03A2 2A14CF                     LD     HL,(CF14H)
03A5 FD7502                     LD     (IY+2),L
03A8 FD7403                     LD     (IY+3),H
03AB CDA204                     CALL   BREAD
03AE CD1B04                     CALL   MOFF
03B1 C30200                     JP     NST
03B4                    ;
03B4                    ;
03B4 21CE02             NODISK: LD     HL,MES9
03B7 1E0A                       LD     E,AH
03B9 0E0D                       LD     C,DH
03BB CD6D02                     CALL   DISPCS
03BE C38105                     JP     ERROR1
03C1                    ; READY CHECK
03C1                    ;
03C1                    READY:  ENT
```

```
03C1 3AE6FF                 LD     A,(MTFG)
03C4 0F                     RRCA
03C5 D4F403                 CALL   NC,MTON
03C8 3AECFF                 LD     A,(DRINO)      ;DRIVE NO GET
03CB F684                   OR     84H
03CD D3DC                   OUT    (DM),A         ;DRIVE SELECT MOTON
03CF AF                     XOR    A
03D0 CD0C06                 CALL   DLY60M
03D3 210000                 LD     HL,00H
03D6 2B             REDY0:  DEC    HL
03D7 7C                     LD     A,H
03D8 B5                     OR     L
03D9 28D9                   JR     Z,NODISK
03DB DBD8                   IN     A,(CR)         ;STATUS GET
03DD 2F                     CPL
03DE 07                     RLCA
03DF 38F5                   JR     C,REDY0
03E1 3AECFF                 LD     A,(DRINO)
03E4 4F                     LD     C,A
03E5 21E7FF                 LD     HL,CLBF0
03E8 0600                   LD     B,00H
03EA 09                     ADD    HL,BC
03EB CB46                   BIT    0,(HL)
03ED C0                     RET    NZ
03EE CD3104                 CALL   RCLB
03F1 CBC6                   SET    0,(HL)
03F3 C9                     RET
03F4                ;
03F4                ; MOTOR ON
03F4                ;
03F4                MTON:   ENT
03F4 3E80                   LD     A,80H
03F6 D3DC                   OUT    (DM),A
03F8 060A                   LD     B,10           ; 1SEC DELAY
03FA 21193C         MTD1:   LD     HL,3C19H
03FD 2B             MTD2:   DEC    HL
03FE 7D                     LD     A,L
03FF B4                     OR     H
0400 20FB                   JR     NZ,MTD2
0402 10F6                   DJNZ   MTD1
0404 3E01                   LD     A,1
0406 32E6FF                 LD     (MTFG),A
0409 C9                     RET
040A                ;
040A                ;SEEK TREATMENT
040A                ;
040A                SEEK:   ENT
040A 3E1B                   LD     A,1BH
040C 2F                     CPL
040D D3D8                   OUT    (CR),A
040F CD4904                 CALL   BUSY
0412 CD0C06                 CALL   DLY60M
0415 DBD8                   IN     A,(CR)
0417 2F                     CPL
0418 E699                   AND    99H
041A C9                     RET
041B                ;
041B                ;MOTOR OFF
041B                ;
041B                MOFF:   ENT
041B CD0506                 CALL   DLY1M          ;1000US DELAY
041E AF                     XOR    A
041F D3DC                   OUT    (DM),A
0421 32E7FF                 LD     (CLBF0),A
0424 32E8FF                 LD     (CLBF1),A
0427 32E9FF                 LD     (CLBF2),A
042A 32EAFF                 LD     (CLBF3),A
042D 32E6FF                 LD     (MTFG),A
0430 C9                     RET
0431
```

```
0431                    ;RECALIBLATION
0431                    ;
0431                    RCLB:   ENT
0431 E5                         PUSH    HL
0432 3E0B                       LD      A,0BH
0434 2F                         CPL
0435 D3D8                       OUT     (CR),A
0437 CD4904                     CALL    BUSY
043A CD0C06                     CALL    DLY60M
043D DBD8                       IN      A,(CR)
043F 2F                         CPL
0440 E685                       AND     85H
0442 EE04                       XOR     04H
0444 E1                         POP     HL
0445 C8                         RET     Z
0446 C37E05                     JP      ERROR
0449                    ;
0449                    ;BUSY AND WAIT
0449                    ;
0449                    BUSY:   ENT
0449 D5                         PUSH    DE
044A E5                         PUSH    HL
044B CDFE05                     CALL    DLY80U
044E 1E07                       LD      E,7
0450 210000             BUSY2:  LD      HL,00H
0453 2B                 BUSY0:  DEC     HL
0454 7C                         LD      A,H
0455 B5                         OR      L
0456 2809                       JR      Z,BUSY1
0458 DBD8                       IN      A,(CR)
045A 2F                         CPL
045B 0F                         RRCA
045C 38F5                       JR      C,BUSY0
045E E1                         POP     HL
045F D1                         POP     DE
0460 C9                         RET
0461 1D                 BUSY1:  DEC     E
0462 20EC                       JR      NZ,BUSY2
0464 C37E05                     JP      ERROR
0467                    ;
0467                    ;DATA CHECK
0467                    ;
0467 0600               CONVRT: LD      B,0
0469 111000                     LD      DE,16
046C 2A1ECF                     LD      HL,(CF1EH)
046F AF                         XOR     A
0470 ED52               TRANS:  SBC     HL,DE
0472 3803                       JR      C,TRANS1
0474 04                         INC     B
0475 18F9                       JR      TRANS
0477 19                 TRANS1: ADD     HL,DE
0478 60                         LD      H,B
0479 2C                         INC     L
047A FD7404                     LD      (IY+4),H
047D FD7505                     LD      (IY+5),L
0480 3AECFF             DCHK:   LD      A,(DRINO)
0483 FE04                       CP      4
0485 3018                       JR      NC,DTCK1
0487 FD7E04                     LD      A,(IY+4)
048A FE46                       CP      70
048C 3011                       JR      NC,DTCK1
048E FD7E05                     LD      A,(IY+5)
0491 B7                         OR      A
0492 280B                       JR      Z,DTCK1
0494 FE11                       CP      17
0496 3007                       JR      NC,DTCK1
0498 FD7E02                     LD      A,(IY+2)
049B FDB603                     OR      (IY+3)
049E C0                         RET     NZ
049F C37E05             DTCK1:  JP      ERROR
```

```
04A2                  ;
04A2                  ;SEQENTIAL READ
04A2                  ;
04A2                  BREAD:  ENT
04A2 F3                       DI
04A3 CD6704                   CALL    CONVRT
04A6 3E0A                     LD      A,10
04A8 32EBFF                   LD      (RETRY),A
04AB CDC103           READ1:  CALL    READY
04AE FD5603                   LD      D,(IY+3)
04B1 FD7E02                   LD      A,(IY+2)
04B4 B7                       OR      A
04B5 2801                     JR      Z,RE0
04B7 14                       INC     D
04B8 FD7E05           RE0:    LD      A,(IY+5)
04BB FD7701                   LD      (IY+1),A
04BE FD7E04                   LD      A,(IY+4)
04C1 FD7700                   LD      (IY+0),A
04C4 DDE5                     PUSH    IX
04C6 E1                       POP     HL
04C9 2F                       CPL
04CA D3DB                     OUT     (DR),A
04CC 3004                     JR      NC,RE1
04CE 3E01                     LD      A,01H
04D0 1802                     JR      RE2
04D2 3E00             RE1:    LD      A,00
04D4 2F               RE2:    CPL
04D5 D3DD                     OUT     (HS),A
04D7 CD0A04                   CALL    SEEK
04DA 206A                     JR      NZ,REE
04DC 0EDB                     LD      C,DBH
04DE FD7E00                   LD      A,(IY+0)
04E1 CB3F                     SRL     A
04E3 2F                       CPL
04E4 D3D9                     OUT     (TR),A
04E6 FD7E01                   LD      A,(IY+1)
04E9 2F                       CPL
04EA D3DA                     OUT     (SCR),A
04EC D9                       EXX
04ED 211F05                   LD      HL,RE3
04F0 E5                       PUSH    HL
04F1 D9                       EXX
04F2 3E94                     LD      A,94H           ;READ & CMD
04F4 2F                       CPL
04F5 D3D8                     OUT     (CR),A
04F7 CD5505                   CALL    WAIT
04FA 0600             RE6:    LD      B,00H
04FC DBD8             RE4:    IN      A,(CR)
04FE 0F                       RRCA
04FF D8                       RET     C
0500 0F                       RRCA
0501 38F9                     JR      C,RE4
0503 EDA2                     INI
0505 20F5                     JR      NZ,RE4
0507 FD3401                   INC     (IY+1)
050A FD7E01                   LD      A,(IY+1)
050D FE11                     CP      17
050F 2805                     JR      Z,RETS
0511 15                       DEC     D
0512 20E6                     JR      NZ,RE6
0514 1801                     JR      RE5
0516 15               RETS:   DEC     D
0517 3ED8             RE5:    LD      A,D8H           ;FORCE INTER RUPT
0519 2F                       CPL
051A D3D8                     OUT     (CR),A
051C CD4904                   CALL    BUSY
051F DBD8             RE3:    IN      A,(CR)
0521 2F                       CPL
```

```
0522 E6FF                 AND     FFH
0524 2020                 JR      NZ,REE
0526 D9                   EXX
0527 E1                   POP     HL
0528 D9                   EXX
0529 FD7E01               LD      A,(IY+1)
052C FE11                 CP      17
052E 2008                 JR      NZ,REX
0530 3E01                 LD      A,01H
0532 FD7701               LD      (IY+1),A
0535 FD3400               INC     (IY+0)
0538 7A          REX:     LD      A,D
0539 B7                   OR      A
053A 2005                 JR      NZ,RE7
053C 3E80                 LD      A,80H
053E D3DC                 OUT     (DM),A
0540 C9                   RET
0541 FD7E00      RE7:     LD      A,(IY+0)
0544 1881                 JR      RE8
0546 3AEBFF      REE:     LD      A,(RETRY)
0549 3D                   DEC     A
054A 32EBFF               LD      (RETRY),A
054D 282F                 JR      Z,ERROR
054F CD3104               CALL    RCLB
0552 C3AB04               JP      READ1
0555             ;
0555             ; WAIT AND BUSY OFF
0555             ;
0555 D5          WAIT:    PUSH    DE
0556 E5                   PUSH    HL
0557 CDFE05               CALL    DLY80U
055A 1E08                 LD      E,8
055C 210000      WAIT2:   LD      HL,00H
055F 2B          WAIT0:   DEC     HL
0560 7C                   LD      A,H
0561 B5                   OR      L
0562 2809                 JR      Z,WAIT1
0564 DBD8                 IN      A,(CR)
0566 2F                   CPL
0567 0F                   RRCA
0568 30F5                 JR      NC,WAIT0
056A E1                   POP     HL
056B D1                   POP     DE
056C C9                   RET
056D 1D          WAIT1:   DEC     E
056E 20EC                 JR      NZ,WAIT2
0570 180C                 JR      ERROR
0572             ;
0572 211603      NMASTE:  LD      HL,MES14
0575 1E07                 LD      E,7H
0577 0E1B                 LD      C,27D
0579 CD6D02               CALL    DISPCS
057C 1803                 JR      ERROR1
057E             ;
057E             ;;;;;;;;;;;;;;;;;;;;;;;;;;;;
057E             ;                          ;
057E             ;    ERROR OR BREAK        ;
057E             ;                          ;
057E             ;;;;;;;;;;;;;;;;;;;;;;;;;;;;
057E CD6602      ERROR:   CALL    BOOTER
0581 CD1B04      ERROR1:  CALL    MOFF
0584 31E0FF      TRYAG2:  LD      SP,FFE0H
0587             ;
0587             ;TRYAG
0587             ;
0587 CDA000      TRYAG:   CALL    FDCC
058A 2047                 JR      NZ,TRYAG3
058C 21DB02               LD      HL,MES10
058F 1E5A                 LD      E,5AH
0591 0E0C                 LD      C,12D
0593 CD6002               CALL    DISP2
```

```
0596 1EAB                  LD      E,ABH
0598 0E11                  LD      C,17D
059A CD6002                CALL    DISP2
059D 1ED3                  LD      E,D3H
059F 0E0F                  LD      C,15D
05A1 CD6002                CALL    DISP2
05A4 CD8C00        TRYAG1: CALL    KEYS1
05A7 CB5F                  BIT     3,A
05A9 CAAC00                JP      Z,CMT
05AC CB77                  BIT     6,A
05AE 2802                  JR      Z,DNO
05B0 18F2                  JR      TRYAG1
05B2 210703        DNO:    LD      HL,MES13          ;DRIVE NO SELECT
05B5 1E0A                  LD      E,AH
05B7 0E0F                  LD      C,FH
05B9 CD6D02                CALL    DISPCS
05BC 1612          DNO10:  LD      D,12H
05BE CDE905                CALL    DNO0
05C1 3009                  JR      NC,DNO3
05C3 1618                  LD      D,18H
05C5 CDE905                CALL    DNO0
05C8 3002                  JR      NC,DNO3
05CA 18F0                  JR      DNO10
05CC 78            DNO3:   LD      A,B
05CD 32ECFF                LD      (DRINO),A
05D0 C36403                JP      FD
05D3               ;
05D3 213103        TRYAG3: LD      HL,MES15
05D6 1E54                  LD      E,54H
05D8 0E1D                  LD      C,29
05DA CD6002                CALL    DISP2
05DD 0606          TRYAG4: LD      B,6
05DF CD8E00        TRYAG5: CALL    KEYS
05E2 CB5F                  BIT     3,A
05E4 CAAC00                JP      Z,CMT
05E7 18F6                  JR      TRYAG5
05E9               ;
05E9 DBE8          DNO0:   IN      A,(E8H)
05EB E6F0                  AND     F0H
05ED B2                    OR      D
05EE D3E8                  OUT     (E8H),A
05F0 DBEA                  IN      A,(EAH)
05F2 0600                  LD      B,0
05F4 0E04                  LD      C,4
05F6 0F                    RRCA
05F7 0F            DNO1:   RRCA
05F8 D0                    RET     NC
05F9 04                    INC     B
05FA 0D                    DEC     C
05FB 20FA                  JR      NZ,DNO1
05FD C9                    RET
05FE               ;
05FE               ;  TIME DELAY (1M &60M &80U )
05FE               ;
05FE D5            DLY80U: PUSH    DE
05FF 110D00                LD      DE,13
0602 C31006                JP      DLYT
0605 D5            DLY1M:  PUSH    DE
0606 118200                LD      DE,130D
0609 C31006                JP      DLYT
060C D5            DLY60M: PUSH    DE
060D 112C1A                LD      DE,6700
0610 1B            DLYT:   DEC     DE
0611 7B                    LD      A,E
0612 B2                    OR      D
0613 20FB                  JR      NZ,DLYT
0615 D1                    POP     DE
0616 C9                    RET
0617               ;
0617               ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
0617                       ;INPUT BUFFER ADDRESS
0617                       ;
0617  P                    IBADR1: EQU    CF00H
0617  P                    IBADR2: EQU    8000H
0617                       ;
0617                       ;   SUBROUTINE WORK
0617                       ;
0617  P                    NTRACK: EQU    FFE0H
0617  P                    NSECT:  EQU    FFE1H
0617  P                    BSIZE:  EQU    FFE2H
0617  P                    STTR:   EQU    FFE4H
0617  P                    STSE:   EQU    FFE5H
0617  P                    MTFG:   EQU    FFE6H
0617  P                    CLBF0:  EQU    FFE7H
0617  P                    CLBF1:  EQU    FFE8H
0617  P                    CLBF2:  EQU    FFE9H
0617  P                    CLBF3:  EQU    FFEAH
0617  P                    RETRY:  EQU    FFEBH
0617  P                    DRINO:  EQU    FFECH
0617                       ;
0617                       ;;;;;;;;;;;;;;;;;;;;
0617                       ;                  ;
0617                       ;   INTRAM¯EXROM   ;
0617                       ;                  ;
0617                       ;;;;;;;;;;;;;;;;;;;;
0617 210080                EXROMT: LD     HL,8000H
061A DD212006                      LD     IX,EROM1
061E 181A                          JR     SEROMA
0620 DBF9                  EROM1:  IN     A,(F9H)
0622 FE00                          CP     00H
0624 C29800                        JP     NZ,NKIN
0627 DD212D06                      LD     IX,EROM2
062B 180D                  EROMT1: JR     SEROMA
062D DBF9                  EROM2:  IN     A,(F9H)
062F 77                            LD     (HL),A
0630 23                            INC    HL
0631 7D                            LD     A,L
0632 B4                            OR     H
0633 20F6                          JR     NZ,EROMT1
0635 D3F8                          OUT    (F8H),A
0637 C30200                        JP     NST
063A                       ;
063A 7C                    SEROMA: LD     A,H
063B D3F8                          OUT    (F8H),A
063D 7D                            LD     A,L
063E D3F9                          OUT    (F9H),A
0640 1604                          LD     D,4
0642 15                    SEROMD: DEC    D
0643 20FD                          JR     NZ,SEROMD
0645 DDE9                          JP     (IX)
0647                       ;
0647                       ;
0647                               END
```

?RDD	0110	?RDI	00E9	ATRB	CF00	BOOTER	0266	BREAD	04A2
BSIZE	FFE2	BUSY	0449	BUSY0	0453	BUSY1	0461	BUSY2	0450
CLBF0	FFE7	CLBF1	FFE8	CLBF2	FFE9	CLBF3	FFEA	CLEAR	0036
CMT	00AC	CONVRT	0467	CR	00D8	D1M	0237	DCHK	0480
DEL1M	024A	DEL6	0244	DISP1	027E	DISP2	0260	DISP3	0277
DISPCS	026D	DLY1M	0605	DLY2	0250	DLY60M	060C	DLY80U	05FE
DLYT	0610	DM	00DC	DNO	05B2	DN00	05E9	DN01	05F7
DN010	05BC	DN03	05CC	DR	00DB	DRINO	FFEC	DTADR	CF14
DTCK1	049F	EDGE	0162	EDGE1	016A	EIRTN	010E	EROM1	0620
EROM2	062D	EROMT1	062B	ERROR	057E	ERROR1	0581	EXROMT	0617
FD	0364	FDCC	00A0	GCLEAR	006E	HS	00DD	IBADR1	CF00
IBADR2	8000	IBUFE	CF00	IPLMC	035D	KEYIN	0060	KEYMES	020E
KEYS	008E	KEYS1	008C	LDMSG	0257	MCHECK	0387	MES1	0289
MES10	02DB	MES11	02E7	MES12	02F8	MES13	0307	MES14	0316
MES15	0331	MES16	034E	MES3	0297	MES6	02B3	MES8	02C1
MES9	02CE	MISMCH	00D9	MOFF	041B	MOT1	01DD	MOTOR	01C7
MOTR	01F2	MOTRD	01EF	MSTOP	01F6	MTD1	03FA	MTD2	03FD
MTFG	FFE6	MTON	03F4	NAME	CF01	NKIN	0098	NMASTE	0572
NODISK	03B4	NSECT	FFE1	NST	0002	NTRACK	FFE0	OPEN	0202
PLAY	0219	RBY1	0177	RBY2	0188	RBY3	0191	RBYTE	0173
RCLB	0431	RD1	00F8	RDDAT	0110	RDINF	00E9	RE0	04B8
RE1	04D2	RE2	04D4	RE3	051F	RE4	04FC	RE5	0517
RE6	04FA	RE7	0541	RE8	04C7	READ1	04AB	READY	03C1
REDY0	03D6	REE	0546	RET2S	0107	RETRY	FFEB	RETS	0516
REW	0225	REX	0538	RTAPE	011C	RTP2	0121	RTP3	0133
RTP5	0154	RTP6	015A	SCR	00DA	SEEK	040A	SEROMA	063A
SEROMD	0642	SIZE	CF12	SPDIN	01BB	ST1	00CE	START	0006
STPEIR	010B	STSE	FFE5	STTR	FFE4	SUMDT	FFE0	TM0	019C
TM1	019F	TM2	01A2	TM3	01AC	TMARK	0193	TMCNT	FFE2
TR	00D9	TRANS	0470	TRANS1	0477	TRTN1	015E	TRYAG	0587
TRYAG1	05A4	TRYAG2	0584	TRYAG3	05D3	TRYAG4	05DD	TRYAG5	05DF
WAIT	0555	WAIT0	055F	WAIT1	056D	WAIT2	055C		

A.4 MONITOR MZ-1Z001Mのプログラムリスト

MONITOR MZ-1Z001Mのプログラム・リストを示します。このプログラム・リストはMZ-80シリーズFDOSのZ80Aアセンブラによるアセブリリストであり、リスト上の各欄の意味は次のようになっています。

ただし、MONITOR MZ-1Z001Mは、先頭番地が＄0000なので、アセンブリ・リスト上の相対アドレスおよびリロケータブルOBJコードは、そのまま絶対アドレスおよびOBJコードと見做すことが可能です。

このプログラム・リストは参考データであり、内容についての問い合せには応じかねます。

```
0000                    ;
0000                    ; 04.15.1982
0000                    ;
0000                    ;  MONITOR MZ-1Z001M
0000                    ;
0000                    ;
0000 C33B00             MONIT:  JP      START           ; RST0
0003                    ;
0003 0000               FLPOS:  DEFW    0000H
0005 00                 ONTYO:  DEFB    00H
0006 00                 CH#:    DEFB    0
0007 00                 TOF:    DEFB    0
0008                    ;
0008 C30000                     JP      MONIT           ; RST1
000B                    ;
000B 01                 SCROST: DEFB    01H
000C 18                 SCREND: DEFB    18H
000D 00                 FLASH:  DEFB    0
000E 00                 FLSDAT: DEFB    0
000F 00                 AMPM:   DEFB    0
0010                    ;
0010 C3B100                     JP      ST              ; RST2
0013                    ;
0013 00D0               SCRST:  DEFW    D000H
0015 00                 SWRK:   DEFB    0
0016 0000               INIC1:  DEFW    0
0018                    ;
0018 C3B100                     JP      ST              ; RST3
001B                    ;
001B 8007               SCRSIZ: DEFW    0780H
001D 00                 TEMPW:  DEFB    0
001E 0000               C2DATA: DEFW    0
0020                    ;
0020 C3B100                     JP      ST              ; RST4
0023                    ;
0023 3030               FOARE:  DEFW    3030H
0025 0D                         DEFB    0DH
0026 00                 KDATW:  DEFB    0
0027 00                 KDATW1: DEFB    0
0028                    ;
0028 C3B100                     JP      ST              ; RST5
002B                    ;
002B                    SHL:    ENT
002B 0000               SUMDT:  DEFW    0
002D 00                 STRGF:  DEFB    0
002E 0000               STACK:  DEFW    0
0030                    ;
0030 C3B100                     JP      ST              ; RST6
0033                    ;
0033                    EHL:    ENT
0033 0000               CSMDT:  DEFW    0
0035 40                 REPTCT: DEFB    40H
0036 0000               RATIO:  DEFW    0
0038                    ;
0038 C3070D                     JP      REGIST          ; RST7
003B                            SKP     H
003B 216900             START:  LD      HL,IOTBL        ;MONITOR COLD START
003E 7E                         LD      A,(HL)
003F 23                         INC     HL
0040 4E                         LD      C,(HL)
0041 23                         INC     HL
0042 ED79                       OUT     (C),A
0044 3C                         INC     A
0045 20F7                       JR      NZ,START+3
0047 314011                     LD      SP,IBUFE
004A CDEB00                     CALL    SYOKI
004D CDEE0C                     CALL    CHR40
0050 062F                       LD      B,2FH
0052 21D011                     LD      HL,KMODE
0055 3E12                       LD      A,12H
```

```
0057 77                     LD      (HL),A
0058 23                     INC     HL
0059 CD4F06                 CALL    ?CLER
005C 3E0D                   LD      A,0DH
005E D3E3                   OUT     (E3H),A
0060 06A0                   LD      B,A0H
0062 00                     NOP
0063 00                     NOP
0064 1829                   JR      START2
0066 C33B00                 JP      START
0069               ;
0069               ;
0069 02E3          IOTBL:   DEFW    E302H         ;8255 CONTROL
006B 34E7                   DEFW    E734H         ;8253 C0 MODE2
006D 74E7                   DEFW    E774H         ;8253 C1 MODE2
006F B4E7                   DEFW    E7B4H         ;8253 C2 MODE2
0071 00E6                   DEFW    E600H         ;8253 C2=0
0073 00E6                   DEFW    E600H
0075 02E5                   DEFW    E502H         ;8253 C1=2
0077 00E5                   DEFW    E500H
0079 02E4                   DEFW    E402H         ;8253 C0=2
007B 00E4                   DEFW    E400H
007D CFE9                   DEFW    E9CFH         ;PIO A MODE3
007F 00E9                   DEFW    E900H         ;     ALL OUTPUT
0081 40E8                   DEFW    E840H
0083 01F7                   DEFW    F701H
0085 00F6                   DEFW    F600H
0087 07F5                   DEFW    F507H
0089 00F4                   DEFW    F400H
008B CFEB                   DEFW    EBCFH         ;PIO B MODE3
008D FFEB                   DEFW    EBFFH         ;     ALL INPUT
008F               ;
008F               ;
008F CD5006        START2:  CALL    ?DINT
0092 210B12                 LD      HL,120BH
0095 22F011                 LD      (FKAE),HL
0098 3E04                   LD      A,4
009A 321D00                 LD      (TEMPW),A
009D 3C                     INC     A
009E 320500                 LD      (ONTYO),A
00A1 119310                 LD      DE,TITMES
00A4 CDB605                 CALL    NLMSG
00A7 ED56                   IM      1
00A9 3EFF          SS:      LD      A,FF
00AB 321500                 LD      (SWRK),A
00AE C3B100        GOOUT:   JP      EXIT
00B1 3E0D          ST:      LD      A,0DH         ;CMT READ MODE
00B3 D3E3                   OUT     (E3H),A
00B5 314011                 LD      SP,IBUFE
00B8 CD290A                 CALL    NL
00BB 3E2A                   LD      A,2AH         ; *
00BD CDC608                 CALL    PRNT
00C0 11AB10                 LD      DE,BUFER
00C3 3E27                   LD      A,39
00C5 CD9107                 CALL    GETLKN
00C8 FE2A                   CP      2AH           ; *
00CA 20E5                   JR      NZ,ST
00CC 13                     INC     DE
00CD 1A                     LD      A,(DE)
00CE FE4A                   CP      'J'
00D0 CA4B02                 JP      Z,JUMP
00D3 FE4D                   CP      'M'
00D5 2822                   JR      Z,MCLECT
00D7 FE44                   CP      'D'
00D9 2845                   JR      Z,DUMP
00DB FE4C                   CP      'L'
00DD CACB01                 JP      Z,MLOAD
00E0 FE53                   CP      'S'
00E2 286A                   JR      Z,MSAVE
00E4 FE56                   CP      'V'
```

```
00E6 CA1702             JP     Z,MVRFY
00E9 18C6               JR     ST
00EB            ;
00EB 3EFF       SYOKI:  LD     A,FFH
00ED D3E0               OUT    (E0H),A
00EF 32B004             LD     (CMODE),A
00F2 210118             LD     HL,1801H
00F5 220B00             LD     (SCROST),HL
00F8 C9                 RET
00F9            ;
00F9            ;
00F9            ;
00F9            ;
00F9 3E4D       MCLECT: LD     A,4DH          ; M
00FB CD8C05             CALL   KIN
00FE 2B                 DEC    HL
00FF 23         MR:     INC    HL
0100 CDBC05             CALL   NLPHLS
0103 7E                 LD     A,(HL)
0104 CDDD05             CALL   PRTHX
0107 CDC408             CALL   PRNTS
010A 11AB10             LD     DE,BUFER
010D CDC505             CALL   GETLBR
0110 11B310             LD     DE,BUFER+8
0113 1A                 LD     A,(DE)
0114 FE0D               CP     0DH            ; CR
0116 28E7               JR     Z,MR
0118 CD2306             CALL   2HEX
011B 38DC               JR     C,MCLECT
011D 77                 LD     (HL),A
011E 18DF               JR     MR
0120            ;
0120            ;
0120            ;
0120 CD7A05     DUMP:   CALL   SSET
0123 CD8305             CALL   ESET
0126 EB                 EX     DE,HL
0127 2A2B00             LD     HL,(SHL)
012A CDBC05     DUMP0:  CALL   NLPHLS
012D 0610               LD     B,16           ; XCHG
012F CDC408     DUMP1:  CALL   PRNTS
0132 7E                 LD     A,(HL)
0133 CDDD05             CALL   PRTHX
0136 E5                 PUSH   HL
0137 AF                 XOR    A
0138 ED52               SBC    HL,DE
013A E1                 POP    HL
013B CAB100             JP     Z,ST
013E CD6205             CALL   BRKEY
0141 28F8               JR     Z,-6
0143 DBEA               IN     A,(EAH)
0145 FEFD               CP     FDH            ; SPKEY
0147 28FA               JR     Z,-4
0149 23                 INC    HL
014A 10E3               DJNZ   DUMP1
014C 18DC               JR     DUMP0
014E            ;
014E            ;
014E            ;
014E 3E02       MSAVE:  LD     A,2
0150 321602             LD     (MWARK),A
0153            ;
0153 21C001     MENAME: LD     HL,FNCOM
0156 11AB10             LD     DE,BUFER
0159 010B00             LD     BC,11
015C D5                 PUSH   DE
015D EDB0               LDIR
015F EB                 EX     DE,HL
0160 0611               LD     B,11H
0162 3E0D               LD     A,0DH
```

```
0164 CD5006              CALL   ?DINT
0167 D1                  POP    DE
0168 CDB605              CALL   NLMSG
016B CDC505              CALL   GETLBR
016E 3A1602              LD     A,(MWARK)
0171 FE02                CP     2
0173 C2CF01              JP     NZ,MLOVE
0176 114011              LD     DE,IBUFE
0179 3E01                LD     A,1
017B 12                  LD     (DE),A
017C 13                  INC    DE
017D 21B510              LD     HL,BUFER+10
0180 011000              LD     BC,16
0183 EDB0                LDIR
0185 3E0D                LD     A,0DH
0187 12                  LD     (DE),A
0188 CD7A05     MNAM1:   CALL   SSET
018B 225411              LD     (DTADR),HL
018E CD8305              CALL   ESET
0191 ED5B2B00            LD     DE,(SHL)
0195 ED52                SBC    HL,DE
0197 38EF                JR     C,MNAM1
0199 23                  INC    HL
019A 225211              LD     (SIZE),HL
019D 21B100              LD     HL,ST
01A0 225611              LD     (EXADR),HL
01A3 3E4A                LD     A,4AH            ; J
01A5 329905              LD     (KINP+1),A
01A8 CD9805     KIN2:    CALL   KINP
01AB 2808                JR     Z,SAVEGO
01AD CD1406              CALL   HLHEX
01B0 38F6                JR     C,KIN2
01B2 225611              LD     (EXADR),HL
01B5 CD5102     SAVEGO:  CALL   ?WRI
01B8 3803                JR     C,+5
01BA CD8202              CALL   ?WRD
01BD C3B100     JST1:    JP     ST
01C0            ;
01C0 46494C45   FNCOM:   DEFM   'FILE NAME:'
01C4 204E414D
01C8 453A
01CA 0D                  DEFB   0DH
01CB            ;
01CB            ;
01CB            ;
01CB 3E01       MLOAD:   LD     A,1
01CD 1881                JR     MSAVE+2
01CF            ;
01CF 21B510     MLOVE:   LD     HL,BUFER+10
01D2 7E                  LD     A,(HL)
01D3 FE0D                CP     0DH
01D5 F5                  PUSH   AF
01D6 CD8E02              CALL   ?RDI
01D9 38E2                JR     C,JST1
01DB F1                  POP    AF
01DC C4F601              CALL   NZ,NAMECK
01DF 20EE                JR     NZ,MLOVE
01E1 3A1602              LD     A,(MWARK)
01E4 3D                  DEC    A
01E5 2033                JR     NZ,MVERY
01E7 112D02              LD     DE,LOAMES
01EA CDCF05              CALL   DSPNAM
01ED CDB202              CALL   ?RDD
01F0 38CB                JR     C,JST1
01F2 2A5611              LD     HL,(EXADR)
01F5 E9                  JP     (HL)
01F6            ;
01F6 114102     NAMECK:  LD     DE,FOUMES
01F9 CDCF05              CALL   DSPNAM
01FC 11B510              LD     DE,BUFER+10
```

```
01FF 214111              LD     HL,NAME
0202 0610                LD     B,16
0204 CD3A06              CALL   SAME
0207 C8                  RET    Z
0208 CDB104              CALL   SERSP
020B CDCE04              CALL   MSTOP
020E 38AD                JR     C,JST1
0210 CD1105              CALL   DEL6
0213 C1                  POP    BC              ; ADJ
0214 18B9                JR     MLOVE
0216            ;
0216            MWARK:   DEFS   1
0217            ;
0217            ;
0217            ;
0217 AF         MVRFY:   XOR    A
0218 18B3                JR     MLOAD+2
021A            ;
021A 113602     MVERY:   LD     DE,VERMES
021D CDCF05              CALL   DSPNAM
0220 CDBE02              CALL   ?VRFY
0223 3898                JR     C,JST1
0225 114802              LD     DE,OKMES
0228 CDB605     NMSGST:  CALL   NLMSG
022B 1890                JR     JST1
022D            ;
022D 4C4F4144   LOAMES:  DEFM   'LOADING '
0231 494E4720
0235 0D                  DEFB   0DH
0236 56455249   VERMES:  DEFM   'VERIFYING '
023A 4659494E
023E 4720
0240 0D                  DEFB   0DH
0241 464F554E   FOUMES:  DEFM   'FOUND '
0245 4420
0247 0D                  DEFB   0DH
0248 4F4B       OKMES:   DEFM   'OK'
024A 0D                  DEFB   0DH
024B            ;
024B            ;
024B            ;
024B 3E4A       JUMP:    LD     A,4AH           ; J
024D CD8C05              CALL   KIN
0250 E9                  JP     (HL)
0251            ;
0251            ;
0251            ;
0251            WRINF:   ENT
0251 F3         ?WRI:    DI
0252 1601                LD     D,1             ; 0001:WI
0254 214011              LD     HL,IBUFE
0257 018000              LD     BC,0080H
025A CD2304     WRI1:    CALL   CKSUM
025D CD5704              CALL   MOTOR
0260 384B                JR     C,STPRET
0262 CB42                BIT    0,D
0264 280B                JR     Z,WRI2          ; WD
0266 D5                  PUSH   DE
0267 117B06              LD     DE,WRIMES
026A CDCF05              CALL   DSPNAM
026D D1                  POP    DE
026E CDF904              CALL   TSPE
0271 CDC703     WRI2:    CALL   GAP
0274 CDDA02              CALL   WTAPE
0277 3834                JR     C,STPRET
0279 CB4A                BIT    1,D
027B C4F904              CALL   NZ,TSPE
027E 202D                JR     NZ,STPRET
0280 FB                  EI
0281 C9                  RET
```

```
0282                  WRDAT:  ENT
0282 F3               ?WRD:   DI
0283 1602                     LD      D,2          ; 0010:WD
0285 ED4B5211                 LD      BC,(SIZE)
0289 2A5411                   LD      HL,(DTADR)
028C 18CC                     JR      WRI1
028E                  ;
028E                  ;
028E                  ;
028E                  RDINF:  ENT
028E F3               ?RDI:   DI
028F 1604                     LD      D,4          ; 0100:RI
0291 214011                   LD      HL,IBUFE
0294 018000                   LD      BC,0080H
0297 CD5704           RD1:    CALL    MOTOR
029A 3811                     JR      C,STPRET
029C CDF103                   CALL    TMARK
029F 380C                     JR      C,STPRET
02A1 CD0B03                   CALL    RTAPE
02A4 3807                     JR      C,STPRET
02A6 CB5A             RD2:    BIT     3,D
02A8 2806                     JR      Z,STPRET+3
02AA CDB104                   CALL    SERSP
02AD CDCE04           STPRET: CALL    MSTOP
02B0 FB                       EI
02B1 C9                       RET
02B2                  RDDAT:  ENT
02B2 F3               ?RDD:   DI
02B3 1608                     LD      D,8          ; 1000:RD
02B5 ED4B5211                 LD      BC,(SIZE)
02B9 2A5411                   LD      HL,(DTADR)
02BC 18D9                     JR      RD1
02BE                  ;
02BE                  ;
02BE                  ;
02BE                  VERFY:  ENT
02BE F3               ?VRFY:  DI
02BF 1608                     LD      D,8          ; RD
02C1 ED4B5211                 LD      BC,(SIZE)
02C5 2A5411                   LD      HL,(DTADR)
02C8 CD2304                   CALL    CKSUM
02CB CD5704                   CALL    MOTOR
02CE 38DD                     JR      C,STPRET
02D0 CDF103                   CALL    TMARK
02D3 38D8                     JR      C,STPRET
02D5 CD5803                   CALL    TVRFY
02D8 18CA                     JR      RD2-2
02DA                  ;
02DA                  ;
02DA                  ;
02DA 1E02             WTAPE:  LD      E,2
02DC C5                       PUSH    BC
02DD E5                       PUSH    HL
02DE 7E               WTAP1:  LD      A,(HL)
02DF CD8F03                   CALL    WBYTE
02E2 CD6C05                   CALL    BRK
02E5 3818                     JR      C,RETHB
02E7 23               WTAP2:  INC     HL
02E8 0B                       DEC     BC
02E9 78                       LD      A,B
02EA B1                       OR      C
02EB 20F1                     JR      NZ,WTAP1
02ED 2A2B00                   LD      HL,(SUMDT)
02F0 7C                       LD      A,H
02F1 CD8F03                   CALL    WBYTE
02F4 7D                       LD      A,L
02F5 CD8F03                   CALL    WBYTE
02F8 CD3905                   CALL    LONG
02FB 1D                       DEC     E
02FC 2004                     JR      NZ,WTAP4
```

```
02FE AF                    XOR    A
02FF E1           RETHB:   POP    HL
0300 C1                    POP    BC
0301 C9                    RET
0302 CD1D05       WTAP4:   CALL   SHORT
0305 10FB                  DJNZ   -3
0307 E1                    POP    HL
0308 C1                    POP    BC
0309 18D1                  JR     WTAPE+2
030B              ;
030B              ;
030B              ;
030B 1E02         RTAPE:   LD     E,2
030D C5                    PUSH   BC
030E E5                    PUSH   HL
030F CD4604       RTAP1:   CALL   EDGE
0312 38EB                  JR     C,RETHB
0314 CD5405                CALL   DLYR
0317 DBE1                  IN     A,(E1H)
0319 E640                  AND    40H
031B 28F2                  JR     Z,RTAP1
031D 210000                LD     HL,0000H
0320 222B00                LD     (SUMDT),HL
0323 E1                    POP    HL
0324 C1                    POP    BC
0325 C5                    PUSH   BC
0326 E5                    PUSH   HL
0327 CDA003       RTAP2:   CALL   RBYTE
032A 38D3                  JR     C,RETHB
032C 77                    LD     (HL),A
032D 23                    INC    HL
032E 0B                    DEC    BC
032F 78                    LD     A,B
0330 B1                    OR     C
0331 20F4                  JR     NZ,RTAP2
0333 2A2B00                LD     HL,(SUMDT)
0336 CDA003                CALL   RBYTE
0339 38C4                  JR     C,RETHB
033B 4F                    LD     C,A
033C CDA003                CALL   RBYTE
033F 38BE                  JR     C,RETHB
0341 BD                    CP     L
0342 2006                  JR     NZ,RTAP3
0344 79                    LD     A,C
0345 BC                    CP     H
0346 3E00                  LD     A,0
0348 28B5                  JR     Z,RETHB
034A 1D           RTAP3:   DEC    E
034B 20C2                  JR     NZ,RTAP1
034D 119206       TAPER:   LD     DE,SUMMES
0350 CDB605                CALL   NLMSG
0353 3EFF                  LD     A,FFH
0355 37                    SCF
0356 18A7                  JR     RETHB
0358              ;
0358              ;
0358              ;
0358 1E02         TVRFY:   LD     E,2
035A C5                    PUSH   BC
035B E5                    PUSH   HL
035C CD4604       TVF1:    CALL   EDGE
035F 389E                  JR     C,RETHB
0361 CD5405                CALL   DLYR
0364 DBE1                  IN     A,(E1H)
0366 E640                  AND    40H
0368 28F2                  JR     Z,TVF1
036A CDA003       TVF2:    CALL   RBYTE
036D 3890                  JR     C,RETHB
036F BE                    CP     (HL)
0370 20DB                  JR     NZ,TAPER
```

```
0372 23                 INC     HL
0373 0B                 DEC     BC
0374 78                 LD      A,B
0375 B1                 OR      C
0376 20F2               JR      NZ,TVF2
0378 2A3300             LD      HL,(CSMDT)
037B CDA003             CALL    RBYTE
037E BC                 CP      H
037F 20CC               JR      NZ,TAPER
0381 CDA003             CALL    RBYTE
0384 BD                 CP      L
0385 20C6               JR      NZ,TAPER
0387 1D                 DEC     E
0388 CAFF02             JP      Z,RETHB
038B E1                 POP     HL
038C C1                 POP     BC
038D 18CB               JR      TVRFY+2
038F            ;
038F            ;
038F            ;
038F C5         WBYTE:  PUSH    BC
0390 0608               LD      B,8
0392 CD3905             CALL    LONG
0395 07         WBY1:   RLCA
0396 DC3905             CALL    C,LONG
0399 D41D05             CALL    NC,SHORT
039C 10F7               DJNZ    WBY1
039E C1                 POP     BC
039F C9                 RET
03A0            ;
03A0            ;
03A0            ;
03A0 E5         RBYTE:  PUSH    HL
03A1 210008             LD      HL,0800H
03A4 CD4604     RBY1:   CALL    EDGE
03A7 381C               JR      C,RBY3
03A9 CD5405             CALL    DLYR
03AC DBE1               IN      A,(E1H)
03AE E640               AND     40H
03B0 280A               JR      Z,RBY2
03B2 E5                 PUSH    HL
03B3 2A2B00             LD      HL,(SUMDT)
03B6 23                 INC     HL
03B7 222B00             LD      (SUMDT),HL
03BA E1                 POP     HL
03BB 37                 SCF
03BC CB15       RBY2:   RL      L
03BE 25                 DEC     H
03BF 20E3               JR      NZ,RBY1
03C1 CD4604             CALL    EDGE
03C4 7D                 LD      A,L
03C5 E1         RBY3:   POP     HL
03C6 C9                 RET
03C7            ;
03C7            ;
03C7            ;
03C7 C5         GAP:    PUSH    BC
03C8 E5                 PUSH    HL
03C9 01F82A             LD      BC,2AF8H
03CC 211414             LD      HL,1414H
03CF CB4A               BIT     1,D
03D1 2004               JR      NZ,GAP1          ; WD
03D3 011027             LD      BC,2710H         ; 55F0H(K)
03D6 29                 ADD     HL,HL
03D7 CD1D05     GAP1:   CALL    SHORT
03DA 0B                 DEC     BC
03DB 78                 LD      A,B
03DC B1                 OR      C
03DD 20F8               JR      NZ,GAP1
03DF CD3905     GAP2:   CALL    LONG
```

```
03E2 25                     DEC     H
03E3 20FA                   JR      NZ,GAP2
03E5 CD1D05         GAP3:   CALL    SHORT
03E8 2D                     DEC     L
03E9 20FA                   JR      NZ,GAP3
03EB CD3905                 CALL    LONG
03EE E1             RETHB1: POP     HL
03EF C1                     POP     BC
03F0 C9                     RET
03F1                ;
03F1                ;
03F1                ;
03F1 E5             TMARK:  PUSH    HL
03F2 2E14                   LD      L,14H
03F4 CB5A                   BIT     3,D
03F6 2002                   JR      NZ,TM1
03F8 CB05                   RLC     L
03FA 65             TM1:    LD      H,L
03FB CD4604         TM2:    CALL    EDGE
03FE 3821                   JR      C,TM4
0400 CD5405                 CALL    DLYR
0403 DBE1                   IN      A,(E1H)
0405 E640                   AND     40H
0407 28F1                   JR      Z,TM1
0409 25                     DEC     H
040A 20EF                   JR      NZ,TM2
040C 65                     LD      H,L
040D CD4604         TM3:    CALL    EDGE
0410 380F                   JR      C,TM4
0412 CD5405                 CALL    DLYR
0415 DBE1                   IN      A,(E1H)
0417 E640                   AND     40H
0419 20DF                   JR      NZ,TM1
041B 25                     DEC     H
041C 20EF                   JR      NZ,TM3
041E CD4604                 CALL    EDGE
0421 E1             TM4:    POP     HL
0422 C9                     RET
0423                ;
0423                ;
0423                ;
0423 C5             CKSUM:  PUSH    BC
0424 E5                     PUSH    HL
0425 D5                     PUSH    DE
0426 110000                 LD      DE,0000H
0429 78             CKS1:   LD      A,B
042A B1                     OR      C
042B 200A                   JR      NZ,CKS2
042D EB                     EX      DE,HL
042E 222B00                 LD      (SUMDT),HL
0431 223300                 LD      (CSMDT),HL
0434 D1                     POP     DE
0435 18B7                   JR      RETHB1
0437 7E             CKS2:   LD      A,(HL)
0438 C5                     PUSH    BC
0439 0608                   LD      B,8
043B 07             CKS3:   RLCA
043C 3001                   JR      NC,+3
043E 13                     INC     DE
043F 10FA                   DJNZ    CKS3
0441 C1                     POP     BC
0442 23                     INC     HL
0443 0B                     DEC     BC
0444 18E3                   JR      CKS1
0446                ;
0446                ;
0446                ;
0446 DBE1           EDGE:   IN      A,(E1H)
0448 2F                     CPL
0449 07                     RLCA
```

```
044A D8                      RET     C
044B 07                      RLCA
044C 30F8                    JR      NC,EDGE
044E DBE1         EDGE1:     IN      A,(E1H)
0450 2F                      CPL
0451 07                      RLCA
0452 D8                      RET     C
0453 07                      RLCA
0454 38F8                    JR      C,EDGE1
0456 C9                      RET
0457              ;
0457              ;
0457              ;
0457 CD7105       MOTOR:     CALL    KBSET
045A DBE1                    IN      A,(E1H)
045C E620                    AND     20H
045E 2818                    JR      Z,MOT2
0460 D5                      PUSH    DE
0461 117206                  LD      DE,SETMES
0464 CDB605                  CALL    NLMSG
0467 D1                      POP     DE
0468 CD8C04                  CALL    OPEN
046B CD6C05       MOT1:      CALL    BRK
046E D8                      RET     C
046F DBE1                    IN      A,(E1H)
0471 E620                    AND     20H
0473 20F6                    JR      NZ,MOT1
0475 CD1705                  CALL    DEL1M
0478 3E03         MOT2:      LD      A,3             ; 0011:WRITE  1100:READ
047A A2                      AND     D
047B 281E                    JR      Z,PLAY
047D DBE1         MOTW:      IN      A,(E1H)
047F E610                    AND     10H
0481 2814                    JR      Z,MOTWG
0483 D5                      PUSH    DE
0484 118406                  LD      DE,WPRMES
0487 CDB605                  CALL    NLMSG
048A D1                      POP     DE
048B 37                      SCF
048C              ;
048C 3E08         OPEN:      LD      A,08H
048E D3E3                    OUT     (E3H),A
0490 CD1705                  CALL    DEL1M
0493 3C                      INC     A
0494 D3E3                    OUT     (E3H),A
0496 C9                      RET
0497              ;
0497 3E0C         MOTWG:     LD      A,0CH           ; WRITE MODE
0499 D3E3                    OUT     (E3H),A
049B 7A           PLAY:      LD      A,D
049C E605                    AND     05H
049E C4D204                  CALL    NZ,MPLAY
04A1 3AB004                  LD      A,(CMODE)
04A4 CB97                    RES     2,A
04A6 182F                    JR      BLK4
04A8                         DEFS    2
04AA              ;
04AA C5           DEL50M:    PUSH    BC
04AB 013100                  LD      BC,49           ;50mS WAIT
04AE 1854                    JR      D1M
04B0              ;
04B0 FF           CMODE:     DEFB    FFH
04B1              ;
04B1              ;
04B1 CDCE04       SERSP:     CALL    MSTOP
04B4 CD1105                  CALL    DEL6
04B7 CDE104                  CALL    FFWD
04BA CD1105                  CALL    DEL6
04BD 01A601       SSP1:      LD      BC,01A6H
04C0 DBE1                    IN      A,(E1H)
```

```
04C2 2F                    CPL
04C3 07                    RLCA
04C4 D8                    RET     C
04C5 07                    RLCA
04C6 30F5                  JR      NC,SSP1
04C8 0B                    DEC     BC
04C9 78                    LD      A,B
04CA B1                    OR      C
04CB 20F3                  JR      NZ,SSP1+3
04CD C9                    RET
04CE              ;
04CE              ;
04CE              ;
04CE              MSTOP:   ENT
04CE 3E0D                  LD      A,0DH           ; READ MODE
04D0 D3E3                  OUT     (E3H),A
04D2 3AB004       MPLAY:   LD      A,(CMODE)
04D5 CB9F                  RES     3,A
04D7 CDDD04       BLK4:    CALL    BLK1
04DA 3AB004       BLK3:    LD      A,(CMODE)
04DD D3E0         BLK1:    OUT     (E0H),A
04DF 1830                  JR      DEL6
04E1              ;
04E1 CD7105       FFWD:    CALL    KBSET
04E4 3AB004                LD      A,(CMODE)
04E7 CBBF                  RES     7,A
04E9 D3E0                  OUT     (E0H),A
04EB CDAA04                CALL    DEL50M
04EE 3AB004                LD      A,(CMODE)
04F1 CB8F                  RES     1,A
04F3 CDDD04                CALL    BLK1
04F6 18E2                  JR      BLK3
04F8 00                    NOP
04F9              ;
04F9              ;
04F9 3E0E         TSPE:    LD      A,0EH
04FB D3E3                  OUT     (E3H),A
04FD CD0005                CALL    DELT
0500              ;
0500              ;
0500              ;
0500 C5           DELT:    PUSH    BC
0501 012B0F                LD      BC,3883         ; 4S WAIT
0504 F5           D1M:     PUSH    AF
0505 AF                    XOR     A
0506 3D                    DEC     A
0507 20FD                  JR      NZ,-1
0509 0B                    DEC     BC
050A 78                    LD      A,B
050B B1                    OR      C
050C 20F7                  JR      NZ,D1M+1
050E F1                    POP     AF
050F C1                    POP     BC
0510 C9                    RET
0511              ;
0511 C5           DEL6:    PUSH    BC
0512 012301                LD      BC,291          ; 300mS WAIT
0515 18ED                  JR      D1M
0517              ;
0517 C5           DEL1M:   PUSH    BC
0518 019607                LD      BC,1942         ; 2S WAIT
051B 18E7                  JR      D1M
051D              ;
051D              ;
051D F5           SHORT:   PUSH    AF
051E 3E0F                  LD      A,0FH
0520 D3E3                  OUT     (E3H),A
0522 0A                    LD      A,(BC)
0523 3E2A                  LD      A,2AH           ; 2AH(H):166.75US
0525 325C05                LD      (DLY+1),A       ; 3FH(L):240.25US
```

```
0528 CD5B05                 CALL   DLY
052B 3E0E                   LD     A,0EH
052D D3E3                   OUT    (E3H),A
052F 3E25                   LD     A,25H          ; 25H(H):166US
0531 325C05                 LD     (DLY+1),A      ; 3AH(L):221.5US
0534 CD5B05                 CALL   DLY
0537 F1                     POP    AF
0538 C9                     RET
0539                ;
0539 F5             LONG:   PUSH   AF
053A 3E0F                   LD     A,0FH
053C D3E3                   OUT    (E3H),A
053E 3E5A                   LD     A,5AH          ; 5AH(H):333US
0540 325C05                 LD     (DLY+1),A      ; 81H(L):469.5US
0543 CD5B05                 CALL   DLY
0546 3E0E                   LD     A,0EH
0548 D3E3                   OUT    (E3H),A
054A 3E55                   LD     A,55H          ; 55H(H):334US
054C 325C05                 LD     (DLY+1),A      ; 7CH(L):452.5US
054F CD5B05                 CALL   DLY
0552 F1                     POP    AF
0553 C9                     RET
0554                ;
0554 7C             DLYR:   LD     A,H
0555 7D                     LD     A,L
0556 3E41                   LD     A,41H          ; 66H(K)
0558 325C05                 LD     (DLY+1),A      ;
055B                ;
055B 3EFF           DLY:    LD     A,FFH
055D 3D                     DEC    A
055E C25D05                 JP     NZ,-1
0561 C9                     RET
0562                ;
0562                ;
0562                BRKEY:  ENT
0562 CD7105                 CALL   KBSET
0565 DBEA                   IN     A,(EAH)
0567 DBEA                   IN     A,(EAH)
0569 E680                   AND    80H
056B C9                     RET
056C                ;
056C                ;
056C                ;
056C DBEA           BRK:    IN     A,(EAH)
056E 2F                     CPL
056F 07                     RLCA
0570 C9                     RET
0571                ;
0571                ;
0571                ;
0571 DBE8           KBSET:  IN     A,(E8H)
0573 E6E0                   AND    E0H
0575 F613                   OR     13H
0577 D3E8                   OUT    (E8H),A
0579 C9                     RET
057A                ;
057A                ;
057A                ;
057A 3E53           SSET:   LD     A,53H          ; S
057C CD8C05                 CALL   KIN
057F 222B00                 LD     (SHL),HL
0582 C9                     RET
0583                ;
0583 3E45           ESET:   LD     A,45H          ; E
0585 CD8C05                 CALL   KIN
0588 223300                 LD     (EHL),HL
058B C9                     RET
058C                ;
058C                ;
058C 329905         KIN:    LD     (KINP+1),A
```

```
058F CD9805          KIN1:   CALL    KINP
0592 CD1406                  CALL    HLHEX
0595 38F8                    JR      C,KIN1
0597 C9                      RET
0598                 ;
0598                 ;
0598                 ;
0598 3EFF            KINP:   LD      A,FFH
059A 11AB10                  LD      DE,BUFER
059D 12                      LD      (DE),A
059E D5                      PUSH    DE
059F 13                      INC     DE
05A0 216B06                  LD      HL,COMES
05A3 010700                  LD      BC,7
05A6 EDB0                    LDIR
05A8 D1                      POP     DE
05A9 CDB605                  CALL    NLMSG
05AC CDC505                  CALL    GETLBR
05AF 11B210                  LD      DE,BUFER+7
05B2 1A                      LD      A,(DE)
05B3 FE0D                    CP      0DH
05B5 C9                      RET
05B6                 ;
05B6                 NLMSG:  ENT
05B6 CD290A                  CALL    NL
05B9 C37B08                  JP      MSGX
05BC                 ;
05BC CD290A          NLPHLS: CALL    NL
05BF CDD805                  CALL    PRTHL
05C2 C3C408                  JP      PRNTS
05C5                 ;
05C5 CDA406          GETLBR: CALL    GETL
05C8 1A                      LD      A,(DE)
05C9 FE0B                    CP      0BH
05CB CAB100                  JP      Z,ST
05CE C9                      RET
05CF                 ;
05CF CDB605          DSPNAM: CALL    NLMSG
05D2 114111                  LD      DE,NAME
05D5 C37B08                  JP      MSGX
05D8                 ;
05D8                 ;
05D8                 ;
05D8 7C              PRTHL:  LD      A,H
05D9 CDDD05                  CALL    PRTHX
05DC 7D                      LD      A,L
05DD                 ;
05DD F5              PRTHX:  PUSH    AF
05DE E6F0                    AND     F0H
05E0 0F                      RRCA
05E1 0F                      RRCA
05E2 0F                      RRCA
05E3 0F                      RRCA
05E4 CDF305                  CALL    ASC
05E7 CDC608                  CALL    PRNT
05EA F1                      POP     AF
05EB E60F                    AND     0F
05ED CDF305                  CALL    ASC
05F0 C3C608                  JP      PRNT
05F3                 ;
05F3                 ;
05F3                 ;
05F3 E60F            ASC:    AND     0FH
05F5 C630                    ADD     A,30H
05F7 FE3A                    CP      3AH
05F9 D8                      RET     C
05FA C607                    ADD     A,07H
05FC C9                      RET
05FD                 ;
05FD                 ;
```

```
05FD                    ;
05FD FE47               HEX:    CP      47H
05FF 3011                       JR      NC,HEXCR
0601 FE41                       CP      41H
0603 300A                       JR      NC,HEX1
0605 FE3A                       CP      3AH
0607 3009                       JR      NC,HEXCR
0609 FE30                       CP      30H
060B D8                         RET     C
060C D630                       SUB     30H
060E C9                         RET
060F D637               HEX1:   SUB     37H
0611 C9                         RET
0612 37                 HEXCR:  SCF
0613 C9                         RET
0614                    ;
0614                    ;
0614                    ;
0614 D5                 HLHEX:  PUSH    DE
0615 CD2306                     CALL    2HEX
0618 3807                       JR      C,HL1
061A 67                         LD      H,A
061B CD2306                     CALL    2HEX
061E 3801                       JR      C,HL1
0620 6F                         LD      L,A
0621 D1                 HL1:    POP     DE
0622 C9                         RET
0623                    ;
0623                    ;
0623                    ;
0623 C5                 2HEX:   PUSH    BC
0624 1A                         LD      A,(DE)
0625 13                         INC     DE
0626 CDFD05                     CALL    HEX
0629 380D                       JR      C,2HEX1
062B 07                         RLCA
062C 07                         RLCA
062D 07                         RLCA
062E 07                         RLCA
062F 4F                         LD      C,A
0630 1A                         LD      A,(DE)
0631 13                         INC     DE
0632 CDFD05                     CALL    HEX
0635 3801                       JR      C,2HEX1
0637 B1                         OR      C
0638 C1                 2HEX1:  POP     BC
0639 C9                         RET
063A                    ;
063A                    ;
063A                    ;
063A C5                 SAME:   PUSH    BC
063B D5                         PUSH    DE
063C E5                         PUSH    HL
063D 1A                 SAME1:  LD      A,(DE)
063E BE                         CP      (HL)
063F 2002                       JR      NZ,SAME2
0641 1004                       DJNZ    SAME3
0643 E1                 SAME2:  POP     HL
0644 D1                         POP     DE
0645 C1                         POP     BC
0646 C9                         RET
0647 FE0D               SAME3:  CP      0DH
0649 28F8                       JR      Z,SAME2
064B 13                         INC     DE
064C 23                         INC     HL
064D 18EE                       JR      SAME1
064F                    ;
064F                    ;
064F                    ;
064F AF                 ?CLER:  XOR     A
```

```
0650                    ;
0650 77                 ?DINT:  LD      (HL),A
0651 23                         INC     HL
0652 10FC                       DJNZ    -2
0654 C9                         RET
0655                    ;
0655                    ;
0655                    ;
0655 A6                 CCURS:  AND     (HL)
0656 77                         LD      (HL),A
0657 A7                         AND     A
0658 3E1F                       LD      A,1FH
065A 2802                       JR      Z,CCURS1
065C 3E93                       LD      A,93H
065E 32C806             CCURS1: LD      (KEYW1+4),A
0661 32D706                     LD      (DYSCSL+1),A
0664 C9                         RET
0665                    ;
0665                    ;
0665 22D111             CURS10: LD      (DSPXY),HL
0668 C3750A                     JP      RETN
066B                    ;
066B                    ;
066B 2D414452           COMES:  DEFM    '-ADR.$'
066F 2E24
0671 0D                         DEFB    0DH
0672 53455420           SETMES: DEFM    'SET TAPE'
0676 54415045
067A 0D                         DEFB    0DH
067B 57524954           WRIMES: DEFM    'WRITING '
067F 494E4720
0683 0D                         DEFB    0DH
0684 57524954           WPRMES: DEFM    'WRITE PROTECT'
0688 45205052
068C 4F544543
0690 54
0691 0D                         DEFB    0DH
0692 43484543           SUMMES: DEFM    'CHECK SUM ERROR'
0696 4B205355
069A 4D204552
069E 524F52
06A1 0D                         DEFB    0DH
06A2                    ;
06A2                    ; GETL KEY
06A2                    ;
06A2                    KNUMBS: DEFS    1
06A3                    KNUMB:  DEFS    1
06A4                    ;
06A4                    GETL:   ENT
06A4 F5                         PUSH    AF
06A5 C5                         PUSH    BC
06A6 E5                         PUSH    HL
06A7 D5                         PUSH    DE
06A8 3AA206                     LD      A,(KNUMBS)
06AB 32A306                     LD      (KNUMB),A
06AE CD290C             KEYW:   CALL    ?PONT
06B1 220300                     LD      (FLPOS),HL
06B4 CD3E0C                     CALL    DSPR
06B7 320D00                     LD      (FLASH),A
06BA 320E00                     LD      (FLSDAT),A
06BD ED732E00           KEYW2:  LD      (STACK),SP
06C1 2A0300                     LD      HL,(FLPOS)
06C4 3A0E00             KEYW1:  LD      A,(FLSDAT)
06C7 FE1F                       CP      1FH
06C9 200B                       JR      NZ,DYSCSL
06CB 3A0D00                     LD      A,(FLASH)
06CE 320E00             KEYW3:  LD      (FLSDAT),A
06D1 CD500C                     CALL    DSPW
06D4 1804                       JR      KEYFL
06D6 3E1F               DYSCSL: LD      A,1FH
```

```
06D8 18F4                    JR      KEYW3
06DA F5          KEYFL:      PUSH    AF
06DB E5                      PUSH    HL
06DC 0640                    LD      B,40H
06DE C5          KYFL1:      PUSH    BC
06DF CD0F09                  CALL    KEYKEY
06E2 FE1E                    CP      1EH            ;NO KEY DATA
06E4 2007                    JR      NZ,KEYDIS
06E6 C1          KYFL2:      POP     BC
06E7 10F5                    DJNZ    KYFL1
06E9 E1                      POP     HL
06EA F1                      POP     AF
06EB 18D7                    JR      KEYW1
06ED F5          KEYDIS:     PUSH    AF
06EE FE01                    CP      01H
06F0 382C                    JR      C,DISPM
06F2 FE05                    CP      05H
06F4 3028                    JR      NC,DISPM
06F6 CB50                    BIT     2,B
06F8 2824                    JR      Z,DISPM
06FA 3E78                    LD      A,78H          ;CURSOL KEY ONLY
06FC 32F711                  LD      (KSTD+3),A
06FF 3A3500                  LD      A,(REPTCT)
0702 3D                      DEC     A
0703 323500                  LD      (REPTCT),A
0706 2811                    JR      Z,REPT
0708 F1                      POP     AF
0709 CD2609      REPT1:      CALL    KEY
070C FE01                    CP      01H
070E 3804                    JR      C,REPT2
0710 FE05                    CP      05H
0712 38D9                    JR      C,KEYDIS
0714 CD2908      REPT2:      CALL    FLASW
0717 1822                    JR      KFINO
0719 3E40        REPT:       LD      A,40H
071B 323500                  LD      (REPTCT),A
071E CD2908      DISPM:      CALL    FLASW
0721 F1                      POP     AF
0722 4F                      LD      C,A
0723 3A1500                  LD      A,(SWRK)
0726 B7                      OR      A
0727 CC140F                  CALL    Z,BELL
072A 79                      LD      A,C
072B E6F0                    AND     F0
072D FE20                    CP      20H
072F 79                      LD      A,C
0730 3812                    JR      C,FUNC
0732 CD4008      GT2:        CALL    ?DSP
0735 21A306                  LD      HL,KNUMB
0738 35                      DEC     (HL)
0739 282E                    JR      Z,GTCR
073B 21AE06      KFINO:      LD      HL,KEYW
073E ED7B2E00                LD      SP,(STACK)
0742 E5                      PUSH    HL
0743 C9                      RET
0744 CB67        FUNC:       BIT     4,A
0746 C2C007                  JP      NZ,FTAB
0749 FE0D                    CP      0DH
074B 281C                    JR      Z,GTCR
074D FE0B                    CP      0BH
074F 2805                    JR      Z,GTBRK
0751 CD390A      GT5:        CALL    ?DPCT
0754 18E5                    JR      KFINO
0756 ED7B2E00    GTBRK:      LD      SP,(STACK)
075A E1                      POP     HL
075B E5                      PUSH    HL
075C 360B                    LD      (HL),0BH       ;BREAK
075E 23                      INC     HL
075F 360D                    LD      (HL),0DH       ;CR
0761 CD2E0A      GETLR:      CALL    LETNL
```

```
0764 D1                 POP    DE
0765 E1                 POP    HL
0766 C1                 POP    BC
0767 F1                 POP    AF
0768 C9                 RET
0769 2AD111    GTCR:    LD     HL,(DSPXY)
076C 5C                 LD     E,H
076D 01                 DEFB   01H           ; LD BC,0028H
076E 2800      CK80:    DEFW   0028H         ; XCHG
0770 CD200C             CALL   MAGA
0773 EB                 EX     DE,HL
0774 2AD111             LD     HL,(DSPXY)
0777 2011               JR     NZ,GETLA
0779 13        GTCR0:   INC    DE
077A 1A                 LD     A,(DE)
077B B7                 OR     A
077C 281B               JR     Z,GETLC
077E 3E28               LD     A,40          ; XCHG
0780 81                 ADD    A,C
0781 4F                 LD     C,A
0782 18F5               JR     GTCR0
0784 25        ADDGA:   DEC    H
0785 3E28               LD     A,40          ; XCHG
0787 81                 ADD    A,C
0788 4F                 LD     C,A
0789 C9                 RET
078A 25        GETLA:   DEC    H
078B 1B                 DEC    DE
078C 3D                 DEC    A
078D 28EA               JR     Z,GTCR0
078F 18F9               JR     GETLA
0791                    ;
0791 32A206    GETLKN:  LD     (KNUMBS),A
0794 CDA406             CALL   GETL
0797 1A                 LD     A,(DE)
0798 C9                 RET
0799                    ;
0799                    ;
0799 2E00      GETLC:   LD     L,00H
079B CD2C0C             CALL   ?PNT1
079E ED7B2E00           LD     SP,(STACK)
07A2 D1                 POP    DE
07A3 D5                 PUSH   DE
07A4 C5                 PUSH   BC
07A5 CD5D0C             CALL   DWLDIR
07A8 C1                 POP    BC
07A9 E1                 POP    HL
07AA E5                 PUSH   HL
07AB 41                 LD     B,C
07AC 7E        GLOP1:   LD     A,(HL)
07AD B7                 OR     A
07AE 2002               JR     NZ,+4
07B0 3E20               LD     A,20H
07B2 77                 LD     (HL),A
07B3 23                 INC    HL
07B4 10F6               DJNZ   GLOP1
07B6 360D      GLOP2:   LD     (HL),0DH
07B8 2B                 DEC    HL
07B9 7E                 LD     A,(HL)
07BA FE20               CP     20H
07BC 28F8               JR     Z,GLOP2
07BE 18A1               JR     GETLR
07C0 FE1B      FTAB:    CP     1BH
07C2 282C               JR     Z,TAB
07C4 FE1A               CP     1AH
07C6 284E               JR     Z,F00
07C8 E60F               AND    0FH           ;00-09  F1-F10
07CA 3C                 INC    A
07CB 47                 LD     B,A
07CC 210012             LD     HL,FARE       ; 1200-129F
```

```
07CF 54                 LD      D,H
07D0 5D                 LD      E,L
07D1 7D                 LD      A,L
07D2 FEA0               CP      A0H             ; FARE END
07D4 2810               JR      Z,+18
07D6 7E                 LD      A,(HL)
07D7 23                 INC     HL
07D8 FE0D               CP      0DH
07DA 20F5               JR      NZ,-9
07DC 10F1               DJNZ    -13
07DE 1A         MRUN:   LD      A,(DE)
07DF FE7F               CP      7FH             ;?CR
07E1 CA6907             JP      Z,GTCR
07E4 FE0D               CP      0DH
07E6 CA3B07             JP      Z,KFIN0
07E9 4F                 LD      C,A
07EA CD9C08             CALL    ?PRT
07ED 13                 INC     DE
07EE 18EE               JR      MRUN
07F0 3E03       TAB:    LD      A,03H
07F2 CD390A             CALL    ?DPCT
07F5 3AD111             LD      A,(DSPXY)
07F8 B7                 OR      A
07F9 CA6907             JP      Z,GTCR
07FC 21C011             LD      HL,TABDAT       ;TAB DATA ARER
07FF 23         TAB1:   INC     HL
0800 3E27               LD      A,39            ; XCHG
0802 BE                 CP      (HL)
0803 DA6907             JP      C,GTCR
0806 3AD111     TAB2:   LD      A,(DSPXY)
0809 96                 SUB     (HL)
080A CA3B07             JP      Z,KFIN0
080D 30F0               JR      NC,TAB1
080F 3E03               LD      A,03H
0811 CD390A             CALL    ?DPCT
0814 18F0               JR      TAB2
0816 21A306     FOO:    LD      HL,KNUMB
0819 35                 DEC     (HL)
081A 2801               JR      Z,+3
081C 35                 DEC     (HL)
081D CA6907             JP      Z,GTCR
0820 112300             LD      DE,FOARE
0823 CD7B08             CALL    MSGX
0826 C33B07             JP      KFIN0
0829 2A0300     FLASW:  LD      HL,(FLPOS)
082C 3A0D00             LD      A,(FLASH)
082F C3500C             JP      DSPW
0832            ;
0832            ;
0832            ;
0832            GETKY:  ENT
0832 C5                 PUSH    BC
0833 D5                 PUSH    DE
0834 E5                 PUSH    HL
0835 CD2609             CALL    KEY
0838 E1                 POP     HL
0839 D1                 POP     DE
083A C1                 POP     BC
083B FE1E               CP      1EH
083D C0                 RET     NZ
083E AF                 XOR     A
083F C9                 RET
0840            ;
0840            ;
0840            ;
0840 F5         ?DSP:   PUSH    AF
0841 C5                 PUSH    BC
0842 D5                 PUSH    DE
0843 E5                 PUSH    HL
0844 CD290C     DSP0:   CALL    ?PONT
```

```
0847 CD500C              CALL   DSPW
084A 2AD111              LD     HL,(DSPXY)
084D 7D                  LD     A,L
084E FE27                CP     39           ; XCHG
0850 2017                JR     NZ,DSP4
0852 5C         DSP1:    LD     E,H
0853 CD200C              CALL   MAGA
0856 23                  INC    HL
0857 3601                LD     (HL),1
0859 B7                  OR     A
085A 280A                JR     Z,DSP3
085C 3602       DSPJR:   LD     (HL),2       ;CHR80:JR DSP3+1 XCHG
085E 3D                  DEC    A
085F 2805                JR     Z,DSP3
0861 3603                LD     (HL),3
0863 3D                  DEC    A
0864 2001                JR     NZ,DSP3+1
0866 23         DSP3:    INC    HL
0867 3600                LD     (HL),0
0869 C3020B     DSP4:    JP     CURSR
086C            ;
086C            ;
086C            ;
086C            PRNTT:   ENT
086C CDC408              CALL   PRNTS
086F 3AF211              LD     A,(DPRNT)
0872 B7                  OR     A
0873 C8                  RET    Z
0874 D60A                SUB    10
0876 38F4                JR     C,-10
0878 20FA                JR     NZ,-4
087A C9                  RET
087B            ;
087B            MSGX:    ENT
087B F5                  PUSH   AF
087C C5                  PUSH   BC
087D D5                  PUSH   DE
087E 1A         MSGX1:   LD     A,(DE)
087F FE0D                CP     0DH
0881 2815                JR     Z,MSG2
0883 CDB508              CALL   PRT3
0886 13                  INC    DE
0887 18F5                JR     MSGX1
0889            ;
0889            MSG:     ENT
0889 F5                  PUSH   AF
088A C5                  PUSH   BC
088B D5                  PUSH   DE
088C 1A         MSG1:    LD     A,(DE)
088D FE0D                CP     0DH
088F 2807                JR     Z,MSG2
0891 4F                  LD     C,A
0892 CD9C08              CALL   ?PRT
0895 13                  INC    DE
0896 18F4                JR     MSG1
0898 D1         MSG2:    POP    DE
0899 C1                  POP    BC
089A F1                  POP    AF
089B C9                  RET
089C            ;
089C 79         ?PRT:    LD     A,C
089D E6F0                AND    F0H
089F 79                  LD     A,C
08A0 2013                JR     NZ,PRT3
08A2 CD390A              CALL   ?DPCT
08A5 FE03                CP     03H          ;CURSR
08A7 280F                JR     Z,PRT4
08A9 FE05                CP     05H          ;HOME
08AB 2803                JR     Z,PRT2
08AD FE06                CP     06H          ;CLRS
```

```
08AF C0                         RET     NZ
08B0 AF              PRT2:      XOR     A
08B1 32F211                     LD      (DPRNT),A
08B4 C9                         RET
08B5 CD4008          PRT3:      CALL    ?DSP
08B8 3AF211          PRT4:      LD      A,(DPRNT)
08BB 3C                         INC     A
08BC FEA0                       CP      160
08BE 38F1                       JR      C,PRT2+1
08C0 D6A0                       SUB     160
08C2 18ED                       JR      PRT2+1
08C4                 ;
08C4                 ;
08C4                 PRNTS:     ENT
08C4 3E20                       LD      A,20H
08C6                 ;
08C6                 PRNT:      ENT
08C6 FE0D                       CP      0DH
08C8 CA2E0A                     JP      Z,LETNL
08CB C5                         PUSH    BC
08CC 4F                         LD      C,A
08CD CD9C08                     CALL    ?PRT
08D0 79                         LD      A,C
08D1 C1                         POP     BC
08D2 C9                         RET
08D3                 ;
08D3                 ; SCROL DATA IN
08D3                 ;   (SCROST)=START LINE
08D3                 ;   (SCREND)=END LINE
08D3                 ;   INITIAL:1  , 24
08D3                 SCRSET: ENT
08D3 F5                         PUSH    AF
08D4 C5                         PUSH    BC
08D5 D5                         PUSH    DE
08D6 E5                         PUSH    HL
08D7 3A0B00                     LD      A,(SCROST)
08DA 47                         LD      B,A
08DB 4F                         LD      C,A
08DC 21D8CF                     LD      HL,SCRN-40          ; XCHG
08DF 112800                     LD      DE,0028H            ;XCHG
08E2 19                         ADD     HL,DE
08E3 10FD                       DJNZ    -1
08E5 221300                     LD      (SCRST),HL
08E8                 ;
08E8 3A0C00                     LD      A,(SCREND)
08EB 3C                         INC     A
08EC 91                         SUB     C
08ED 47                         LD      B,A
08EE 210000                     LD      HL,0000H
08F1 19                         ADD     HL,DE
08F2 10FD                       DJNZ    -1
08F4 221B00                     LD      (SCRSIZ),HL
08F7 3E06                       LD      A,06H
08F9 CD390A                     CALL    ?DPCT
08FC E1                         POP     HL
08FD D1                         POP     DE
08FE C1                         POP     BC
08FF F1                         POP     AF
0900 C9                         RET
0901                 ;
0901 21F311          NOKKEY:    LD      HL,KYBDA
0904 77                         LD      (HL),A
0905 3D                         DEC     A
0906 060B                       LD      B,11
0908 23                         INC     HL
0909 77                         LD      (HL),A
090A 10FC                       DJNZ    -2
090C 3E1E                       LD      A,1EH
090E C9                         RET
090F                 ;
```

```
090F                  KEYKEY: ENT
090F  DBE8                    IN      A,(E8H)
0911  CBA7                    RES     4,A
0913  D3E8                    OUT     (E8H),A
0915  DBEA                    IN      A,(EAH)
0917  DBEA                    IN      A,(EAH)
0919  3C                      INC     A
091A  F5                      PUSH    AF
091B  0605                    LD      B,5
091D  AF                      XOR     A
091E  3D                      DEC     A
091F  20FD                    JR      NZ,-1
0921  10FA                    DJNZ    -4
0923  F1                      POP     AF
0924  28DB                    JR      Z,NOKKEY
0926                  ;
0926                  KEY:    ENT
0926  DBE8            KSWEP:  IN      A,(E8H)
0928  E6F0                    AND     F0H
092A  F61B                    OR      1BH
092C  57                      LD      D,A               ;D=STROB
092D  D3E8                    OUT     (E8H),A
092F  AF                      XOR     A
0930  322600                  LD      (KDATW),A
0933  322700                  LD      (KDATW1),A
0936  DBEA                    IN      A,(EAH)
0938  2F                      CPL
0939  47                      LD      B,A               ;B=BIT DATA
093A  0EEA                    LD      C,EAH             ;C=I/O PORT
093C  0D              SWEP:   DEC     C
093D  0D                      DEC     C
093E  15                      DEC     D
093F  ED51                    OUT     (C),D
0941  0C                      INC     C
0942  0C                      INC     C
0943  D5                      PUSH    DE
0944  21F411                  LD      HL,KSTD
0947  7A                      LD      A,D
0948  E60F                    AND     0FH
094A  5F                      LD      E,A
094B  1600                    LD      D,0
094D  19                      ADD     HL,DE
094E  ED78                    IN      A,(C)
0950  5F                      LD      E,A
0951  2F                      CPL
0952  A6                      AND     (HL)
0953  73                      LD      (HL),E
0954  D1                      POP     DE
0955  5F                      LD      E,A
0956  B7                      OR      A
0957  C4240A                  CALL    NZ,DATA1
095A  7A                      LD      A,D               ;STROB END?
095B  E60F                    AND     0FH
095D  20DD                    JR      NZ,SWEP
095F  ED5B2600                LD      DE,(KDATW)
0963  7B                      LD      A,E
0964  B7                      OR      A
0965  202A                    JR      NZ,DATA           ;KSWEP END
0967                  NOKD:   ENT
0967  21F311          NOKD1:  LD      HL,KYBDA          ;SPECIAL BIT DATA
096A  78                      LD      A,B
096B  BE                      CP      (HL)
096C  2820                    JR      Z,KFINA
096E  77                      LD      (HL),A
096F  21D011          NOKD2:  LD      HL,KMODE          ;KGSX XXXX
0972  FE01                    CP      01H
0974  2815                    JR      Z,GRPHO           ;G
0976  FE02                    CP      02H
0978  280E                    JR      Z,SMALLO          ;SL
097A  FE03                    CP      03H               ;DISP CR
```

```
097C 2807               JR      Z,CRDIS         ;G+SL
097E FE08               CP      08H
0980 200C               JR      NZ,KFINA
0982 3E0C      KANA0:   LD      A,0CH
0984 01                 DEFB    1
0985 3E7F      CRDIS:   LD      A,7FH
0987 01                 DEFB    1
0988 3E0A      SMALL0:  LD      A,0AH
098A 01                 DEFB    1
098B 3E09      GRPH0:   LD      A,09H
098D 01                 DEFB    1
098E 3E1E      KFINA:   LD      A,1EH
0990 C9                 RET
0991           ;
0991 7B        DATA:    LD      A,E
0992 1E00               LD      E,00H
0994 B7                 OR      A
0995 1F        ROT:     RRA
0996 3803               JR      C,ROTE
0998 1C                 INC     E
0999 18FA               JR      ROT
099B B7        ROTE:    OR      A
099C 20F0               JR      NZ,KFINA
099E 3E0F      KDIN:    LD      A,0FH
09A0 A2                 AND     D               ;D=STROB DATA
09A1 1F                 RRA
09A2 57                 LD      D,A
09A3 3004               JR      NC,KD1
09A5 3E08               LD      A,08H
09A7 83                 ADD     A,E
09A8 5F                 LD      E,A
09A9 7A        KD1:     LD      A,D
09AA 4A                 LD      C,D             ;D=ADD STROB
09AB 07                 RLCA
09AC 07                 RLCA
09AD 07                 RLCA
09AE 07                 RLCA
09AF B3                 OR      E
09B0 5F                 LD      E,A
09B1 AF                 XOR     A
09B2 57                 LD      D,A
09B3 79                 LD      A,C
09B4 FE02               CP      02H
09B6 3814               JR      C,LMONLY
09B8 21D011             LD      HL,KMODE
09BB 78                 LD      A,B
09BC FE04               CP      04H             ; S
09BE 2832               JR      Z,SHIFT
09C0 CB7E      NOMLK:   BIT     7,(HL)          ; K
09C2 200E               JR      NZ,KKANA
09C4 CB76               BIT     6,(HL)          ; G
09C6 200F               JR      NZ,KGRP
09C8 CB6E               BIT     5,(HL)          ; S
09CA 2021               JR      NZ,KSML
09CC 211E0D    LMONLY:  LD      HL,KTBL
09CF 19        KADD:    ADD     HL,DE
09D0 7E        KATM:    LD      A,(HL)
09D1 C9                 RET
09D2 21AE0D    KKANA:   LD      HL,KTBLK-32
09D5 18F8               JR      KADD
09D7 218E0D    KGRP:    LD      HL,KTBLG-32
09DA 7B                 LD      A,E
09DB FE40               CP      40H
09DD 38F0               JR      C,KADD
09DF FE52               CP      52H
09E1 2807               JR      Z,+9
09E3 FE53               CP      53H
09E5 20A7               JR      NZ,KFINA
09E7 3E07               LD      A,07H
09E9 01                 DEFB    1
```

```
09EA 3E05                   LD      A,05H
09EC C9                     RET
09ED 21560D         KSML:   LD      HL,KTBLS-32
09F0 18DD                   JR      KADD
09F2                ;
09F2 7B             SHIFT:  LD      A,E
09F3 FE52                   CP      52H
09F5 2807                   JR      Z,+9
09F7 FE53                   CP      53H
09F9 2006                   JR      NZ,SHIFT1
09FB 3E08                   LD      A,08H
09FD 01                     DEFB    1
09FE 3E06                   LD      A,06H
0A00 C9                     RET
0A01 CB7E           SHIFT1: BIT     7,(HL)
0A03 200D                   JR      NZ,KSKANA
0A05 CB76                   BIT     6,(HL)
0A07 28E4                   JR      Z,KSML
0A09 21E60D         RVGRP:  LD      HL,KTBLR-32
0A0C FE4A                   CP      4AH
0A0E 300E                   JR      NC,KFINB
0A10 18BD                   JR      KADD
0A12 21360E         KSKANA: LD      HL,KTBLKS
0A15 060D                   LD      B,13
0A17 BE             KSKA1:  CP      (HL)
0A18 2807                   JR      Z,KSKA2
0A1A 23                     INC     HL
0A1B 23                     INC     HL
0A1C 10F9                   DJNZ    KSKA1
0A1E C38E09         KFINB:  JP      KFINA
0A21 23             KSKA2:  INC     HL
0A22 7E                     LD      A,(HL)
0A23 C9                     RET
0A24                ;
0A24                ;
0A24 ED532600       DATA1:  LD      (KDATW),DE
0A28 C9                     RET
0A29                ;
0A29                ;
0A29                ;
0A29                NL:     ENT
0A29 3AF211                 LD      A,(DPRNT)
0A2C B7                     OR      A
0A2D C8                     RET     Z
0A2E                ;
0A2E                LETNL:  ENT
0A2E AF                     XOR     A
0A2F 32F211                 LD      (DPRNT),A
0A32 3E0E                   LD      A,0EH           ; KEY NOMAL MODE
0A34 CD390A                 CALL    ?DPCT
0A37 3E0D                   LD      A,0DH
0A39                ;
0A39                ;
0A39                ;
0A39                ?DPCT:  ENT
0A39 F5                     PUSH    AF
0A3A C5                     PUSH    BC
0A3B D5                     PUSH    DE
0A3C E5                     PUSH    HL
0A3D B7                     OR      A
0A3E 2835                   JR      Z,RETN
0A40 214E0A                 LD      HL,TDPCT
0A43 3D                     DEC     A
0A44 07                     RLCA
0A45 4F                     LD      C,A
0A46 0600                   LD      B,0
0A48 09                     ADD     HL,BC
0A49 5E                     LD      E,(HL)
0A4A 23                     INC     HL
0A4B 56                     LD      D,(HL)
```

```
0A4C EB                 EX     DE,HL
0A4D E9                 JP     (HL)
0A4E E90A       TDPCT:  DEFW   CURSD
0A50 F90A               DEFW   CURSU
0A52 020B               DEFW   CURSR
0A54 1D0B               DEFW   CURSL
0A56 520B               DEFW   HOME
0A58 310B               DEFW   CLRS
0A5A 5A0B               DEFW   DEL
0A5C A10B               DEFW   INST
0A5E 980A               DEFW   GRAPH
0A60 890A               DEFW   SMALL
0A62 750A               DEFW   RETN          ;BREAK
0A64 7A0A               DEFW   KANA
0A66 E80B               DEFW   CR
0A68 6C0A               DEFW   LAMODE
0A6A 6C0A               DEFW   CANRVS
0A6C            ;
0A6C            LAMODE: ENT
0A6C 21D011     CANRVS: LD     HL,KMODE
0A6F 3E00               LD     A,00H
0A71 CD5506     OUTRT:  CALL   CCURS
0A74            ;
0A74            ;
0A74 00                 NOP
0A75 E1         RETN:   POP    HL
0A76 D1                 POP    DE
0A77 C1                 POP    BC
0A78 F1                 POP    AF
0A79 C9                 RET
0A7A 21D011     KANA:   LD     HL,KMODE
0A7D CB7E               BIT    7,(HL)
0A7F CBBE               RES    7,(HL)
0A81 2002               JR     NZ,+4
0A83 CBFE               SET    7,(HL)
0A85 3E9F               LD     A,9FH
0A87 18E8               JR     OUTRT
0A89 21D011     SMALL:  LD     HL,KMODE
0A8C CB6E               BIT    5,(HL)
0A8E CBAE               RES    5,(HL)
0A90 2002               JR     NZ,+4
0A92 CBEE               SET    5,(HL)
0A94 3E3F               LD     A,3FH
0A96 18D9               JR     OUTRT
0A98 21D011     GRAPH:  LD     HL,KMODE
0A9B CB76               BIT    6,(HL)
0A9D CBB6               RES    6,(HL)
0A9F 2002               JR     NZ,+4
0AA1 CBF6               SET    6,(HL)
0AA3 3E5F               LD     A,5FH
0AA5 18CA               JR     OUTRT
0AA7            ;
0AA7            ;
0AA7            ;
0AA7 ED5B1300   SCROL:  LD     DE,(SCRST)
0AAB ED4B1B00           LD     BC,(SCRSIZ)
0AAF 212800             LD     HL,0028H      ; XCHG
0AB2 19                 ADD    HL,DE
0AB3 CD5D0C             CALL   DWLDIR
0AB6 EB                 EX     DE,HL
0AB7 0628               LD     B,40          ; XCHG
0AB9 CD6C0C             CALL   DSCL
0ABC 3A0B00             LD     A,(SCROST)
0ABF 4F                 LD     C,A
0AC0 3A0C00             LD     A,(SCREND)
0AC3 91                 SUB    C
0AC4 C603               ADD    A,03H
0AC6 4F                 LD     C,A
0AC7 0600               LD     B,0
0AC9 11D311             LD     DE,MANG
```

```
0ACC 3A0B00             LD     A,(SCROST)
0ACF 6F                 LD     L,A
0AD0 2600               LD     H,0
0AD2 19                 ADD    HL,DE
0AD3 E5                 PUSH   HL
0AD4 D1                 POP    DE
0AD5 1B                 DEC    DE
0AD6 D5                 PUSH   DE
0AD7 EDB0               LDIR
0AD9 3600               LD     (HL),00H
0ADB E1                 POP    HL
0ADC 7E                 LD     A,(HL)
0ADD B7                 OR     A
0ADE 2895               JR     Z,RETN
0AE0 2AD111     SCRO2:  LD     HL,(DSPXY)
0AE3 25                 DEC    H
0AE4 22D111             LD     (DSPXY),HL
0AE7 18BE               JR     SCROL
0AE9            ;
0AE9 2AD111     CURSD:  LD     HL,(DSPXY)
0AEC 3A0C00             LD     A,(SCREND)
0AEF BC                 CP     H
0AF0 38B5               JR     C,SCROL
0AF2 28B3               JR     Z,SCROL
0AF4 24                 INC    H
0AF5 C36506     CURS1:  JP     CURS10
0AF8 00                 NOP
0AF9 CD120C     CURSU:  CALL   SCRDSD
0AFC D2750A             JP     NC,RETN
0AFF 25                 DEC    H
0B00 18F3               JR     CURS1
0B02            ;
0B02 2AD111     CURSR:  LD     HL,(DSPXY)
0B05 7D                 LD     A,L
0B06 FE27               CP     39            ; XCHG
0B08 3003               JR     NC,CURS2
0B0A 2C                 INC    L
0B0B 18E8               JR     CURS1
0B0D 2E00       CURS2:  LD     L,0
0B0F 24                 INC    H
0B10 3A0C00             LD     A,(SCREND)
0B13 BC                 CP     H
0B14 30DF               JR     NC,CURS1
0B16 67                 LD     H,A
0B17 22D111             LD     (DSPXY),HL
0B1A C3A70A             JP     SCROL
0B1D            ;
0B1D 2AD111     CURSL:  LD     HL,(DSPXY)
0B20 7D                 LD     A,L
0B21 B7                 OR     A
0B22 2803               JR     Z,CURL1
0B24 2D                 DEC    L
0B25 18CE               JR     CURS1
0B27 2E27       CURL1:  LD     L,39          ; XCHG
0B29 CD150C             CALL   SCRSTD
0B2C 3024               JR     NC,HOME
0B2E 25                 DEC    H
0B2F 18C4               JR     CURS1
0B31            ;
0B31            ;
0B31 210B00     CLRS:   LD     HL,SCROST
0B34 5E                 LD     E,(HL)
0B35 23                 INC    HL
0B36 7E                 LD     A,(HL)
0B37 93                 SUB    E
0B38 3C                 INC    A
0B39 3C                 INC    A
0B3A 4F                 LD     C,A
0B3B 2A1300             LD     HL,(SCRST)
0B3E 0628       CLRS1:  LD     B,40          ; XCHG
```

```
0B40 CD6C0C          CALL  DSCL
0B43 0D              DEC   C
0B44 20F8            JR    NZ,CLRS1
0B46 21D311          LD    HL,MANG
0B49 4B              LD    C,E
0B4A 09              ADD   HL,BC
0B4B 3EEF            LD    A,EFH         ; MANG END
0B4D 95              SUB   L
0B4E 47              LD    B,A
0B4F CD4F06          CALL  ?CLER
0B52         ;
0B52 CD150C  HOME:   CALL  SCRSTD
0B55 67              LD    H,A
0B56 2E00            LD    L,0
0B58 189B            JR    CURS1
0B5A         ;
0B5A CD120C  DEL:    CALL  SCRDSD
0B5D 2005            JR    NZ,DEL0
0B5F AF              XOR   A
0B60 B5              OR    L
0B61 CA750A          JP    Z,RETN
0B64 7D      DEL0:   LD    A,L
0B65 B7              OR    A
0B66 2010            JR    NZ,DEL1
0B68 5C              LD    E,H
0B69 CD200C          CALL  MAGA
0B6C 200A            JR    NZ,DEL1
0B6E CD290C          CALL  ?PONT
0B71 2B              DEC   HL
0B72 AF              XOR   A
0B73 CD500C          CALL  DSPW
0B76 18A5            JR    CURSL
0B78 CD1B0C  DEL1:   CALL  DSMAG
0B7B 0600            LD    B,0
0B7D 78      DEL10:  LD    A,B
0B7E C628            ADD   A,40          ; XCHG
0B80 47              LD    B,A
0B81 7E              LD    A,(HL)
0B82 B7              OR    A
0B83 2803            JR    Z,DEL2
0B85 23              INC   HL
0B86 18F5            JR    DEL10
0B88 2AD111  DEL2:   LD    HL,(DSPXY)
0B8B 78              LD    A,B
0B8C 95              SUB   L
0B8D 4F              LD    C,A
0B8E 0600            LD    B,0
0B90 CD290C          CALL  ?PONT
0B93 E5              PUSH  HL
0B94 D1              POP   DE
0B95 1B              DEC   DE
0B96 CD5D0C          CALL  DWLDIR
0B99 2B              DEC   HL
0B9A AF              XOR   A
0B9B CD500C          CALL  DSPW
0B9E C31D0B          JP    CURSL
0BA1 CD1B0C  INST:   CALL  DSMAG
0BA4 EB              EX    DE,HL
0BA5 0E00            LD    C,0
0BA7 2AD111          LD    HL,(DSPXY)
0BAA 2E27            LD    L,39          ; XCHG
0BAC 280A            JR    Z,INST0
0BAE 24      INSTR0: INC   H
0BAF 79              LD    A,C
0BB0 C628            ADD   A,40          ; XCHG
0BB2 4F              LD    C,A
0BB3 13              INC   DE
0BB4 1A              LD    A,(DE)
0BB5 B7              OR    A
0BB6 20F6            JR    NZ,INSTR0
```

```
0BB8 CD2C0C        INST0:  CALL   ?PNT1
0BBB CD3E0C                CALL   DSPR
0BBE B7                    OR     A
0BBF C2750A                JP     NZ,RETN
0BC2 E5                    PUSH   HL
0BC3 2AD111                LD     HL,(DSPXY)
0BC6 3E27                  LD     A,39           ; XCHG
0BC8 95                    SUB    L
0BC9 81                    ADD    A,C
0BCA 47                    LD     B,A
0BCB D1            INST2:  POP    DE
0BCC D5                    PUSH   DE
0BCD E1                    POP    HL
0BCE 2B                    DEC    HL
0BCF DBE8                  IN     A,(E8H)
0BD1 CBFF                  SET    7,A
0BD3 F3                    DI
0BD4 D3E8                  OUT    (E8H),A
0BD6 7E            INST1:  LD     A,(HL)
0BD7 12                    LD     (DE),A
0BD8 3620                  LD     (HL),20H
0BDA 2B                    DEC    HL
0BDB 1B                    DEC    DE
0BDC 10F8                  DJNZ   INST1
0BDE DBE8                  IN     A,(E8H)
0BE0 CBBF                  RES    7,A
0BE2 D3E8                  OUT    (E8H),A
0BE4 FB                    EI
0BE5 C3750A                JP     RETN
0BE8               ;
0BE8 CD1B0C        CR:     CALL   DSMAG
0BEB EB                    EX     DE,HL
0BEC 2AD111                LD     HL,(DSPXY)
0BEF CA0D0B                JP     Z,CURS2
0BF2 0600                  LD     B,0
0BF4 04            CR0:    INC    B
0BF5 13                    INC    DE
0BF6 1A                    LD     A,(DE)
0BF7 B7                    OR     A
0BF8 20FA                  JR     NZ,CR0
0BFA 24            CR1:    INC    H
0BFB 2E00                  LD     L,0
0BFD 3A0C00                LD     A,(SCREND)
0C00 57                    LD     D,A
0C01 7C                    LD     A,H
0C02 80                    ADD    A,B
0C03 BA                    CP     D
0C04 3806                  JR     C,CR2
0C06 22D111                LD     (DSPXY),HL
0C09 C3A70A                JP     SCROL
0C0C 24            CR2:    INC    H
0C0D 10FD                  DJNZ   CR2
0C0F C3F50A                JP     CURS1
0C12               ;
0C12 2AD111        SCRDSD: LD     HL,(DSPXY)
0C15 3A0B00        SCRSTD: LD     A,(SCROST)
0C18 3D                    DEC    A
0C19 BC                    CP     H
0C1A C9                    RET
0C1B               ;
0C1B 2AD111        DSMAG:  LD     HL,(DSPXY)
0C1E 5C                    LD     E,H
0C1F 1C                    INC    E
0C20 1600          MAGA:   LD     D,0
0C22 21D311                LD     HL,MANG
0C25 19                    ADD    HL,DE
0C26 7E                    LD     A,(HL)
0C27 B7                    OR     A
0C28 C9                    RET
0C29               ;
```

```
0C29 2AD111          ?PONT:   LD     HL,(DSPXY)
0C2C C5              ?PNT1:   PUSH   BC
0C2D D5                       PUSH   DE
0C2E E5                       PUSH   HL
0C2F C1                       POP    BC
0C30 04                       INC    B
0C31 112800                   LD     DE,0028H          ; XCHG
0C34 21D8CF                   LD     HL,SCRN-40        ; XCHG
0C37 19              PON1:    ADD    HL,DE
0C38 10FD                     DJNZ   PON1
0C3A 09                       ADD    HL,BC
0C3B D1                       POP    DE
0C3C C1                       POP    BC
0C3D C9                       RET
0C3E                 ;
0C3E                 DSPRED:  ENT
0C3E F3              DSPR:    DI
0C3F C5                       PUSH   BC
0C40 0EE8                     LD     C,E8H
0C42 ED40                     IN     B,(C)
0C44 CBF8                     SET    7,B
0C46 ED41                     OUT    (C),B
0C48 7E                       LD     A,(HL)
0C49 CBB8            DSPWRR:  RES    7,B
0C4B ED41                     OUT    (C),B
0C4D C1                       POP    BC
0C4E FB                       EI
0C4F C9                       RET
0C50 F3              DSPW:    DI
0C51 C5                       PUSH   BC
0C52 0EE8                     LD     C,E8H
0C54 ED40                     IN     B,(C)
0C56 CBF8                     SET    7,B
0C58 ED41                     OUT    (C),B
0C5A 77                       LD     (HL),A
0C5B 18EC                     JR     DSPWRR
0C5D F3              DWLDIR:  DI
0C5E DBE8                     IN     A,(E8H)
0C60 CBFF                     SET    7,A
0C62 D3E8                     OUT    (E8H),A
0C64 EDB0                     LDIR
0C66 CBBF            DWLDRN:  RES    7,A
0C68 D3E8                     OUT    (E8H),A
0C6A FB                       EI
0C6B C9                       RET
0C6C F3              DSCL:    DI
0C6D DBE8                     IN     A,(E8H)
0C6F CBFF                     SET    7,A
0C71 D3E8                     OUT    (E8H),A
0C73 AF                       XOR    A
0C74 77                       LD     (HL),A
0C75 23                       INC    HL
0C76 10FC                     DJNZ   -2
0C78 DBE8                     IN     A,(E8H)
0C7A 18EA                     JR     DWLDRN
0C7C                 ;
0C7C                 ;
0C7C                 ;
0C7C                 CHR80:   ENT
0C7C 3E10                     LD     A,10H
0C7E 322E01                   LD     (DUMP0+4),A
0C81 211809                   LD     HL,0918H
0C84 225C08                   LD     (DSPJR),HL
0C87 3EB0                     LD     A,B0H
0C89 32970C                   LD     (CHX0+1),A
0C8C 3EEF                     LD     A,EFH
0C8E 32D40C                   LD     (CHX2+1),A
0C91 3E4F                     LD     A,4FH
0C93 329F0C                   LD     (CHX1+1),A
0C96 3EB0            CHX0:    LD     A,B0H
```

```
0C98 32DD08            LD      (SCRSET+10),A
0C9B 32350C            LD      (?PNT1+9),A
0C9E 3E4F      CHX1:   LD      A,4FH
0CA0 320108            LD      (TAB1+2),A
0CA3 324F08            LD      (DSPO+11),A
0CA6 32070B            LD      (CURSR+5),A
0CA9 32280B            LD      (CURL1+1),A
0CAC 32AB0B            LD      (INST+10),A
0CAF 32C70B            LD      (INSTO+15),A
0CB2 3C                INC     A
0CB3 326E07            LD      (GTCR+5),A
0CB6 327F07            LD      (GTCRO+6),A
0CB9 328607            LD      (ADDGA+2),A
0CBC 32E008            LD      (SCRSET+13),A
0CBF 32B00A            LD      (SCROL+9),A
0CC2 32B80A            LD      (SCROL+17),A
0CC5 323F0B            LD      (CLRS1+1),A
0CC8 327F0B            LD      (DEL10+2),A
0CCB 32B10B            LD      (INSTRO+3),A
0CCE 32320C            LD      (?PNT1+6),A
0CD1 DBE8              IN      A,(E8H)
0CD3 CBEF      CHX2:   SET     5,A
0CD5 D3E8              OUT     (E8H),A
0CD7 2100D0            LD      HL,D000H
0CDA AF                XOR     A
0CDB D3F6              OUT     (F6H),A         ;GRAPHIC 00
0CDD 47                LD      B,A
0CDE CD6C0C    DCL:    CALL    DSCL
0CE1 7C                LD      A,H
0CE2 FEE0              CP      E0H
0CE4 20F8              JR      NZ,DCL
0CE6 CDD308            CALL    SCRSET
0CE9 3E06              LD      A,06H           ;CL
0CEB C3390A            JP      ?DPCT
0CEE           ;
0CEE           CHR40:  ENT
0CEE 3E08              LD      A,08H
0CF0 322E01            LD      (DUMPO+4),A
0CF3 213602            LD      HL,0236H
0CF6 225C08            LD      (DSPJR),HL
0CF9 3ED8              LD      A,D8H
0CFB 32970C            LD      (CHXO+1),A
0CFE 3EAF              LD      A,AFH
0D00 32D40C            LD      (CHX2+1),A
0D03 3E27              LD      A,27H
0D05 188C              JR      CHXO-3
0D07           ;
0D07           ;
0D07           ;
0D07 E5        REGIST: PUSH    HL
0D08 D5                PUSH    DE
0D09 C5                PUSH    BC
0D0A F5                PUSH    AF
0D0B 0604              LD      B,4
0D0D E1                POP     HL
0D0E CDD805            CALL    PRTHL
0D11 CDC408            CALL    PRNTS
0D14 10F7              DJNZ    -7
0D16 E1                POP     HL
0D17 2B                DEC     HL
0D18 CDD805            CALL    PRTHL
0D1B C3B100            JP      ST
0D1E           ;
0D1E           ;
0D1E           ;KEY TABL
0D1E           ;
0D1E           ;SO
0D1E 1011      KTBL:   DEFW    1110H
0D20 1213              DEFW    1312H
0D22 1415              DEFW    1514H
```

```
0D24 1617                DEFW    1716H
0D26            ;S1
0D26 1819                DEFW    1918H
0D28 3839                DEFW    3938H
0D2A 1A2E                DEFW    2E1AH
0D2C 2B2D                DEFW    2D2BH
0D2E            ;S2
0D2E 3031                DEFW    3130H
0D30 3233                DEFW    3332H
0D32 3435                DEFW    3534H
0D34 3637                DEFW    3736H
0D36            ;S3
0D36 1B20                DEFW    201BH
0D38 0D02                DEFW    020DH
0D3A 0104                DEFW    0401H
0D3C 030B                DEFW    0B03H
0D3E            ;S4
0D3E 2F41                DEFW    412FH
0D40 4243                DEFW    4342H
0D42 4445                DEFW    4544H
0D44 4647                DEFW    4746H
0D46            ;S5
0D46 4849                DEFW    4948H
0D48 4A4B                DEFW    4B4AH
0D4A 4C4D                DEFW    4D4CH
0D4C 4E4F                DEFW    4F4EH
0D4E            ;S6
0D4E 5051                DEFW    5150H
0D50 5253                DEFW    5352H
0D52 5455                DEFW    5554H
0D54 5657                DEFW    5756H
0D56            ;S7
0D56 5859                DEFW    5958H
0D58 5A5E                DEFW    5E5AH
0D5A 5C3F                DEFW    3F5CH
0D5C 2E2C                DEFW    2C2EH
0D5E            ;S8
0D5E 3031                DEFW    3130H
0D60 3233                DEFW    3332H
0D62 3435                DEFW    3534H
0D64 3637                DEFW    3736H
0D66            ;S9
0D66 3839                DEFW    3938H
0D68 3A3B                DEFW    3B3AH
0D6A 2D40                DEFW    402DH
0D6C 5B00                DEFW    005BH
0D6E            ;S10
0D6E 5D00                DEFW    005DH
0D70 0507                DEFW    0705H
0D72 0000                DEFW    0000H
0D74 0000                DEFW    0000H
0D76            KTBLS:   ENT
0D76            ;S4S
0D76 8461                DEFW    6184H
0D78 6263                DEFW    6362H
0D7A 6465                DEFW    6564H
0D7C 6667                DEFW    6766H
0D7E            ;S5S
0D7E 6869                DEFW    6968H
0D80 6A6B                DEFW    6B6AH
0D82 6C6D                DEFW    6D6CH
0D84 6E6F                DEFW    6F6EH
0D86            ;S6S
0D86 7071                DEFW    7170H
0D88 7273                DEFW    7372H
0D8A 7475                DEFW    7574H
0D8C 7677                DEFW    7776H
0D8E            ;S7S
0D8E 7879                DEFW    7978H
0D90 7A7E                DEFW    7E7AH
```

```
0D92 7C82                  DEFW   827CH
0D94 3E3C                  DEFW   3C3EH
0D96              ;S8S
0D96 5F21                  DEFW   215FH
0D98 2223                  DEFW   2322H
0D9A 2425                  DEFW   2524H
0D9C 2627                  DEFW   2726H
0D9E              ;S9S
0D9E 2829                  DEFW   2928H
0DA0 2A2B                  DEFW   2B2AH
0DA2 3D60                  DEFW   603DH
0DA4 7B00                  DEFW   007BH
0DA6              ;S10S
0DA6 7D00                  DEFW   007DH
0DA8 0507                  DEFW   0705H
0DAA 0000                  DEFW   0000H
0DAC 0000                  DEFW   0000H
0DAE              KTBLG:   ENT
0DAE              ;S4G
0DAE 8398                  DEFW   9883H
0DB0 8886                  DEFW   8688H
0DB2 9A9E                  DEFW   9E9AH
0DB4 9B99                  DEFW   999BH
0DB6              ;S5G
0DB6 8089                  DEFW   8980H
0DB8 908D                  DEFW   8D90H
0DBA 8F91                  DEFW   918FH
0DBC 948C                  DEFW   8C94H
0DBE              ;S6G
0DBE 8B97                  DEFW   978BH
0DC0 9F96                  DEFW   969FH
0DC2 9C8A                  DEFW   8A9CH
0DC4 8795                  DEFW   9587H
0DC6              ;S7G
0DC6 859D                  DEFW   9D85H
0DC8 8E1E                  DEFW   1E8EH
0DCA 1E81                  DEFW   811EH
0DCC 1EFF                  DEFW   FF1EH
0DCE              ;
0DCE              KTBLK:   ENT
0DCE D2C1                  DEFW   C1D2H
0DD0 BABF                  DEFW   BFBAH
0DD2 BCB2                  DEFW   B2BCH
0DD4 CAB7                  DEFW   B7CAH
0DD6              ;
0DD6 B8C6                  DEFW   C6B8H
0DD8 CFC9                  DEFW   C9CFH
0DDA D8D3                  DEFW   D3D8H
0DDC D0D7                  DEFW   D7D0H
0DDE              ;
0DDE BEC0                  DEFW   C0BEH
0DE0 BDC4                  DEFW   C4BDH
0DE2 B6C5                  DEFW   C5B6H
0DE4 CBC3                  DEFW   C3CBH
0DE6              ;
0DE6 BBDD                  DEFW   DDBBH
0DE8 C2CD                  DEFW   CDC2H
0DEA A6DB                  DEFW   DBA6H
0DEC D9C8                  DEFW   C8D9H
0DEE              ;
0DEE DCC7                  DEFW   C7DCH
0DF0 CCB1                  DEFW   B1CCH
0DF2 B3B4                  DEFW   B4B3H
0DF4 B5D4                  DEFW   D4B5H
0DF6              ;
0DF6 D5D6                  DEFW   D6D5H
0DF8 B9DA                  DEFW   DAB9H
0DFA CEDE                  DEFW   DECEH
0DFC DF00                  DEFW   00DFH
0DFE              ;
```

```
0DFE D100              DEFW    00D1H
0E00 0507              DEFW    0705H
0E02 0000              DEFW    0000H
0E04 0000              DEFW    0000H
0E06            ;
0E06            KTBLR:  ENT
0E06 00E1              DEFW    E100H
0E08 E2E3              DEFW    E3E2H
0E0A E4E5              DEFW    E5E4H
0E0C E6E7              DEFW    E7E6H
0E0E            ;
0E0E E8E9              DEFW    E9E8H
0E10 EAEB              DEFW    EBEAH
0E12 ECED              DEFW    EDECH
0E14 EEEF              DEFW    EFEEH
0E16            ;
0E16 FAFB              DEFW    FBFAH
0E18 FCFD              DEFW    FDFCH
0E1A FE92              DEFW    92FEH
0E1C A0A4              DEFW    A4A0H
0E1E            ;
0E1E A5B0              DEFW    B0A5H
0E20 E000              DEFW    00E0H
0E22 0000              DEFW    0000H
0E24 0000              DEFW    0000H
0E26            ;
0E26 F0F1              DEFW    F1F0H
0E28 F2F3              DEFW    F3F2H
0E2A F4F5              DEFW    F5F4H
0E2C F6F7              DEFW    F7F6H
0E2E            ;
0E2E F8F9              DEFW    F9F8H
0E30 0000              DEFW    0000H
0E32 0000              DEFW    0000H
0E34 0000              DEFW    0000H
0E36            ;
0E36            KTBLKS: ENT
0E36 25A8              DEFW    A825H
0E38 3AAF              DEFW    AF3AH
0E3A 3EA1              DEFW    A13EH
0E3C 43A7              DEFW    A743H
0E3E 44A9              DEFW    A944H
0E40            ;
0E40 45AA              DEFW    AA45H
0E42 46AB              DEFW    AB46H
0E44 47AC              DEFW    AC47H
0E46 48AD              DEFW    AD48H
0E48 49AE              DEFW    AE49H
0E4A            ;
0E4A 4EA2              DEFW    A24EH
0E4C 50A3              DEFW    A350H
0E4E 40A6              DEFW    A640H
0E50            ;
0E50            ;
0E50            ;TEMPO
0E50            ;      A=VALUE
0E50            XTEMP:  ENT
0E50 F5                PUSH    AF
0E51 C5                PUSH    BC
0E52 E60F              AND     0FH
0E54 47                LD      B,A
0E55 3E08              LD      A,8
0E57 90                SUB     B
0E58 321D00            LD      (TEMPW),A
0E5B C1                POP     BC
0E5C F1                POP     AF
0E5D C9                RET
0E5E            ;
0E5E            ;TIME SET
0E5E            ;      BC=C2
```

```
0E5E                    ;       DE=SECOND
0E5E                    ;       C2=0-FFFF 12H
0E5E                    ;       C1=A8C0H=12HSEC
0E5E                    ;       C0=7A12H=31.25KHZ
0E5E                    TIMST:  ENT
0E5E C5                 ?TMST:  PUSH    BC
0E5F 320F00                     LD      (AMPM),A
0E62 ED531600                   LD      (INIC1),DE
0E66 3EC1                       LD      A,C1H           ;C1=A8C1 SET
0E68 D3E5                       OUT     (E5H),A
0E6A 3EA8                       LD      A,A8H
0E6C D3E5                       OUT     (E5H),A
0E6E 3E02                       LD      A,02H           ;C0=0002 SET
0E70 D3E4                       OUT     (E4H),A
0E72 AF                         XOR     A
0E73 D3E4                       OUT     (E4H),A
0E75                    ;
0E75 D3F0                       OUT     (F0H),A         ;C0 C1 RESET
0E77                    ;
0E77 3E44               TMS1:   LD      A,44H           ;C1 LATCH
0E79 D3E7                       OUT     (E7H),A
0E7B DBE5                       IN      A,(E5H)         ;C1 READ
0E7D 4F                         LD      C,A
0E7E DBE5                       IN      A,(E5H)
0E80 FEA8                       CP      A8H
0E82 20F3                       JR      NZ,TMS1
0E84 3EC1                       LD      A,C1H
0E86 B9                         CP      C
0E87 20EE                       JR      NZ,TMS1
0E89 3EC0                       LD      A,C0H           ;C1=A8C0 SET
0E8B D3E5                       OUT     (E5H),A
0E8D 3EA8                       LD      A,A8H
0E8F D3E5                       OUT     (E5H),A
0E91 3E12                       LD      A,12H           ;C0=7A12 SET
0E93 D3E4                       OUT     (E4H),A
0E95 3E7A                       LD      A,7AH
0E97 D3E4                       OUT     (E4H),A
0E99 3E84                       LD      A,84H           ;C2 LATCH
0E9B D3E7                       OUT     (E7H),A
0E9D DBE6                       IN      A,(E6H)         ;C2 READ
0E9F 4F                         LD      C,A
0EA0 DBE6                       IN      A,(E6H)
0EA2 47                         LD      B,A
0EA3 ED431E00                   LD      (C2DATA),BC
0EA7 C1                         POP     BC
0EA8 C9                         RET
0EA9                    ;
0EA9                    ;TIME READ
0EA9                    ;       BC=C2 12H
0EA9                    ;       DE=SECOND
0EA9                    TIMRD:  ENT
0EA9 C5                 ?TMRD:  PUSH    BC
0EAA E5                         PUSH    HL
0EAB 3E84                       LD      A,84H           ;C2 LATCH
0EAD D3E7                       OUT     (E7H),A
0EAF 3E44                       LD      A,44H           ;C1 LATCH
0EB1 D3E7                       OUT     (E7H),A
0EB3 DBE6                       IN      A,(E6H)         ;C2 READ
0EB5 4F                         LD      C,A
0EB6 DBE6                       IN      A,(E6H)
0EB8 47                         LD      B,A
0EB9 DBE5                       IN      A,(E5H)         ;C1 READ
0EBB 5F                         LD      E,A
0EBC DBE5                       IN      A,(E5H)
0EBE 57                         LD      D,A
0EBF 2A1E00                     LD      HL,(C2DATA)
0EC2 AF                         XOR     A
0EC3 ED42                       SBC     HL,BC
0EC5 7D                         LD      A,L
0EC6 0F                         RRCA
```

```
0EC7 DC070F                 CALL   C,TMUP
0ECA D5                     PUSH   DE
0ECB 7A                     LD     A,D
0ECC B3                     OR     E
0ECD 2003                   JR     NZ,TMX
0ECF 11C0A8                 LD     DE,A8C0H
0ED2 21C0A8         TMX:    LD     HL,A8C0H       ; HL=A8C0-C1
0ED5 ED52                   SBC    HL,DE
0ED7 ED5B1600               LD     DE,(INIC1)     ; HL=HL+INISET
0EDB 19                     ADD    HL,DE
0EDC 3823                   JR     C,TMX1
0EDE E5                     PUSH   HL
0EDF 11C0A8                 LD     DE,A8C0H
0EE2 ED52                   SBC    HL,DE
0EE4 3814                   JR     C,TMR1
0EE6 F1                     POP    AF             ; ADJ
0EE7 EB             TMX2:   EX     DE,HL
0EE8 3A0F00                 LD     A,(AMPM)
0EEB EE01                   XOR    01H
0EED E1                     POP    HL
0EEE 010100                 LD     BC,0001H
0EF1 ED42                   SBC    HL,BC
0EF3 2002                   JR     NZ,+4
0EF5 EE01                   XOR    01H
0EF7 E1                     POP    HL
0EF8 C1                     POP    BC
0EF9 C9                     RET
0EFA D1             TMR1:   POP    DE
0EFB E1                     POP    HL
0EFC 3A0F00                 LD     A,(AMPM)
0EFF 18F6                   JR     -8
0F01 114057         TMX1:   LD     DE,5740H
0F04 19                     ADD    HL,DE
0F05 18E0                   JR     TMX2
0F07 ED431E00       TMUP:   LD     (C2DATA),BC
0F0B 3A0F00                 LD     A,(AMPM)
0F0E EE01                   XOR    01H
0F10 320F00                 LD     (AMPM),A
0F13 C9                     RET
0F14                ;
0F14                ;  BELL
0F14                ;
0F14                BELL:   ENT
0F14 C5                     PUSH   BC
0F15 E5                     PUSH   HL
0F16 013000                 LD     BC,0030H
0F19 216000                 LD     HL,0060H
0F1C CD220F                 CALL   SOUT
0F1F E1                     POP    HL
0F20 C1                     POP    BC
0F21 C9                     RET
0F22                ;
0F22                ; SOUND OUT
0F22                ;          BC=ONCHOO
0F22                ;          HL=ONTEI
0F22 C5             SOUT:   PUSH   BC
0F23 D5                     PUSH   DE
0F24 3E05           SOUT1:  LD     A,05H
0F26 CD350F                 CALL   SOUT2
0F29 3E04                   LD     A,04H
0F2B CD350F                 CALL   SOUT2
0F2E 0B                     DEC    BC
0F2F 79                     LD     A,C
0F30 B0                     OR     B
0F31 20F1                   JR     NZ,SOUT1
0F33 1852                   JR     PORET1
0F35                ;
0F35 D3E3           SOUT2:  OUT    (E3H),A
0F37 54                     LD     D,H
0F38 5D                     LD     E,L
```

```
0F39 1B              DEC   DE
0F3A 7A              LD    A,D
0F3B B3              OR    E
0F3C 20FB            JR    NZ,-3
0F3E C9              RET
0F3F         ;
0F3F         ; MELODY
0F3F         ;         DE=DATA LOW ADDRESS
0F3F         MELDY:  ENT
0F3F C5              PUSH  BC
0F40 D5              PUSH  DE
0F41 E5              PUSH  HL
0F42 3E02            LD    A,2
0F44 32EF11          LD    (OCTV),A
0F47 1A      MLD1:   LD    A,(DE)
0F48 FE0D            CP    0DH
0F4A 283A            JR    Z,MLD4
0F4C FE2A            CP    2AH                  ;* END MARK
0F4E 2836            JR    Z,MLD4
0F50 FE2D            CP    2DH                  ;- UNDER OCTAVE
0F52 2826            JR    Z,MLD2
0F54 FE2B            CP    2BH                  ;+ UPPER OCTAVE
0F56 282A            JR    Z,MLD3
0F58 21EA0F          LD    HL,MTBL
0F5B FE23            CP    23H                  ;#
0F5D 3E00            LD    A,00
0F5F 2005            JR    NZ,+7
0F61 210210          LD    HL,M#TBL
0F64 3C              INC   A
0F65 13              INC   DE
0F66 320600          LD    (CH#),A
0F69 CD8A0F          CALL  ONPU
0F6C 38D9            JR    C,MLD1
0F6E CD3310          CALL  RYTHM
0F71 3E02            LD    A,2
0F73 32EF11          LD    (OCTV),A
0F76 380E            JR    C,MLD4
0F78 18CD            JR    MLD1
0F7A 3E03    MLD2:   LD    A,3
0F7C 32EF11          LD    (OCTV),A
0F7F 13              INC   DE
0F80 18C5            JR    MLD1
0F82 3E01    MLD3:   LD    A,1
0F84 18F6            JR    MLD2+2
0F86 E1      MLD4:   POP   HL
0F87 D1      PORET1: POP   DE
0F88 C1              POP   BC
0F89 C9              RET
0F8A         ;
0F8A         ; ONPU TO RATIO CONV
0F8A         ;         (RATIO)=ONTEI
0F8A         ;         C=ONCHOO*TEMPO
0F8A C5      ONPU:   PUSH  BC
0F8B 0608            LD    B,8
0F8D 1A      ONP1:   LD    A,(DE)
0F8E BE              CP    (HL)
0F8F 2809            JR    Z,ONP2
0F91 23              INC   HL
0F92 23              INC   HL
0F93 23              INC   HL
0F94 10F8            DJNZ  ONP1+1
0F96 37              SCF
0F97 13              INC   DE
0F98 C1              POP   BC
0F99 C9              RET
0F9A 78      ONP2:   LD    A,B
0F9B 320700          LD    (TOF),A
0F9E 23              INC   HL
0F9F D5              PUSH  DE
0FA0 5E              LD    E,(HL)
```

```
0FA1 23                 INC     HL
0FA2 56                 LD      D,(HL)
0FA3 EB                 EX      DE,HL
0FA4 7C                 LD      A,H
0FA5 B5                 OR      L
0FA6 280A               JR      Z,ONP3
0FA8 3AEF11             LD      A,(OCTV)
0FAB 3D                 DEC     A
0FAC 2836               JR      Z,HOCT
0FAE 3D                 DEC     A
0FAF 2801               JR      Z,ONP3
0FB1 29                 ADD     HL,HL
0FB2 223600     ONP3:   LD      (RATIO),HL
0FB5 D1                 POP     DE
0FB6 13                 INC     DE
0FB7 1A                 LD      A,(DE)
0FB8 47                 LD      B,A
0FB9 E6F0               AND     F0H
0FBB FE30               CP      30H
0FBD 2805               JR      Z,+7
0FBF 3A0500             LD      A,(ONTYO)
0FC2 1807               JR      +9
0FC4 13                 INC     DE
0FC5 78                 LD      A,B
0FC6 E60F               AND     0FH
0FC8 320500             LD      (ONTYO),A
0FCB 4F                 LD      C,A
0FCC 0600               LD      B,0
0FCE 211A10             LD      HL,OPTBL
0FD1 09                 ADD     HL,BC
0FD2 D5                 PUSH    DE
0FD3 5E                 LD      E,(HL)
0FD4 50                 LD      D,B
0FD5 3A1D00             LD      A,(TEMPW)
0FD8 47                 LD      B,A
0FD9 62                 LD      H,D
0FDA 6A                 LD      L,D
0FDB 19                 ADD     HL,DE
0FDC 10FD               DJNZ    -1
0FDE D1                 POP     DE
0FDF C1                 POP     BC
0FE0 E5                 PUSH    HL
0FE1 C1                 POP     BC
0FE2 AF                 XOR     A
0FE3 C9                 RET
0FE4 CB3C       HOCT:   SRL     H
0FE6 CB1D               RR      L
0FE8 18C8               JR      ONP3
0FEA            ;
0FEA 43         MTBL:   DEFB    'C'
0FEB 2501               DEFW    0125H
0FED 44                 DEFB    'D'
0FEE 0501               DEFW    0105H
0FF0 45                 DEFB    'E'
0FF1 E900               DEFW    00E9H
0FF3 46                 DEFB    'F'
0FF4 DC00               DEFW    00DCH
0FF6 47                 DEFB    'G'
0FF7 C300               DEFW    00C3H
0FF9 41                 DEFB    'A'
0FFA AE00               DEFW    00AEH
0FFC 42                 DEFB    'B'
0FFD 9B00               DEFW    009BH
0FFF 52                 DEFB    'R'
1000 0000               DEFW    0000H
1002 43         M#TBL:  DEFB    'C'
1003 1501               DEFW    0115H
1005 44                 DEFB    'D'
1006 F600               DEFW    00F6H
1008 45                 DEFB    'E'
```

```
1009 DC00                   DEFW    00DCH
100B 46                     DEFB    'F'
100C CF00                   DEFW    00CFH
100E 47                     DEFB    'G'
100F B800                   DEFW    00B8H
1011 41                     DEFB    'A'
1012 A400                   DEFW    00A4H
1014 42                     DEFB    'B'
1015 9200                   DEFW    0092H
1017 52                     DEFB    'R'
1018 0000                   DEFW    0000H
101A 01           OPTBL:    DEFB    01H
101B 02                     DEFB    02H
101C 03                     DEFB    03H
101D 04                     DEFB    04H
101E 06                     DEFB    06H
101F 08                     DEFB    08H
1020 0C                     DEFB    0CH
1021 10                     DEFB    10H
1022 18                     DEFB    18H
1023 20                     DEFB    20H
1024              ;
1024 08           TABLE1:   DEFB    8
1025 0F                     DEFB    15
1026 0D                     DEFB    13
1027 0C                     DEFB    12
1028 0B                     DEFB    11
1029 0A                     DEFB    10
102A 09                     DEFB    9
102B 08                     DEFB    8
102C 10                     DEFB    16
102D 0E                     DEFB    14
102E 0D                     DEFB    13
102F 0B                     DEFB    11
1030 0B                     DEFB    11
1031 0A                     DEFB    10
1032 08                     DEFB    8
1033              ;
1033              ; RHYTHM
1033              ;
1033              RYTHM:    ENT
1033 CD7105                 CALL    KBSET
1036 CD6C05                 CALL    BRK
1039 D8                     RET     C
103A D5                     PUSH    DE
103B C5                     PUSH    BC
103C C5                     PUSH    BC
103D 212310                 LD      HL,TABLE1-1
1040 3A0600                 LD      A,(CH#)
1043 FE00                   CP      0
1045 2804                   JR      Z,RYTHM1
1047 010700                 LD      BC,7
104A 09                     ADD     HL,BC
104B 3A0700       RYTHM1:   LD      A,(TOF)
104E 4F                     LD      C,A
104F FE01                   CP      1
1051 2005                   JR      NZ,RYTHM3
1053 3E02                   LD      A,2
1055 32EF11                 LD      (OCTV),A
1058 09           RYTHM3:   ADD     HL,BC
1059 46                     LD      B,(HL)
105A 3AEF11                 LD      A,(OCTV)
105D 3D                     DEC     A
105E 2807                   JR      Z,RYTHM2
1060 3D                     DEC     A
1061 2806                   JR      Z,*N
1063 CB38                   SRL     B
1065 1802                   JR      *N
1067 CB20         RYTHM2:   SLA     B
1069 D1           *N:       POP     DE
```

```
106A 210000                 LD     HL,0000H
106D 19                     ADD    HL,DE
106E 10FD                   DJNZ   -1
1070 44                     LD     B,H
1071 4D                     LD     C,L
1072 2A3600                 LD     HL,(RATIO)
1075 7C                     LD     A,H
1076 B5                     OR     L
1077 2806                   JR     Z,RDEL
1079 CD220F                 CALL   SOUT
107C C1             *N1:    POP    BC
107D D1                     POP    DE
107E C9                     RET
107F E5             RDEL:   PUSH   HL
1080 3E04                   LD     A,4
1082 32250F                 LD     (SOUT1+1),A
1085 212501                 LD     HL,0125H
1088 CD220F                 CALL   SOUT
108B 3E05                   LD     A,5
108D 32250F                 LD     (SOUT1+1),A
1090 E1                     POP    HL
1091 18E9                   JR     *N1
1093                ;
1093 2A2A204D ≒     TITMES: DEFM   '** MONITOR MZ-1Z001M **
1097 4F4E4954
109B 4F52204D
109F 5A2D315A
10A3 3030314D
10A7 202A2A
10AA 0D                     DEFB   0DH
10AB                ;
10AB                ;
10AB                BUFER:  DEFS   80
10FB                ;
10FB                ;
10FB  P             IBUFE:  EQU    1140H
10FB  P             ATRB:   EQU    1140H
10FB  P             NAME:   EQU    1141H
10FB  P             SIZE:   EQU    1152H
10FB  P             DTADR:  EQU    1154H
10FB  P             EXADR:  EQU    1156H
10FB  P             COMNT:  EQU    1158H
10FB  P             TABDAT: EQU    11C0H
10FB  P             KMODE:  EQU    11D0H
10FB  P             DSPXY:  EQU    11D1H
10FB  P             MANG:   EQU    11D3H
10FB  P             OCTV:   EQU    11EFH
10FB  P             FKAE:   EQU    11F0H
10FB  P             DPRNT:  EQU    11F2H
10FB  P             KYBDA:  EQU    11F3H
10FB  P             KSTD:   EQU    11F4H
10FB  P             FARE:   EQU    1200H
10FB  P             SCRN:   EQU    D000H
10FB  P             EXIT:   EQU    ST
10FB                ;
10FB                ;
10FB                        END
```

*N	1069	*N1	107C	2HEX	0623	2HEX1	0638	?CLER	064F
?DINT	0650	?DPCT	0A39	?DSP	0840	?PNT1	0C2C	?PONT	0C29
?PRT	089C	?RDD	02B2	?RDI	028E	?TMRD	0EA9	?TMST	0E5E
?VRFY	02BE	?WRD	0282	?WRI	0251	ADDGA	0784	AMPM	000F
ASC	05F3	ATRB	1140	BELL	0F14	BLK1	04DD	BLK3	04DA
BLK4	04D7	BRK	056C	BRKEY	0562	BUFER	10AB	C2DATA	001E
CANRVS	0A6C	CCURS	0655	CCURS1	065E	CH#	0006	CHR40	0CEE
CHR80	0C7C	CHX0	0C96	CHX1	0C9E	CHX2	0CD3	CK80	076E
CKS1	0429	CKS2	0437	CKS3	043B	CKSUM	0423	CLRS	0B31
CLRS1	0B3E	CMODE	04B0	COMES	066B	COMNT	1158	CR	0BE8
CR0	0BF4	CR1	0BFA	CR2	0C0C	CRDIS	0985	CSMDT	0033
CURL1	0B27	CURS1	0AF5	CURS10	0665	CURS2	0B0D	CURSD	0AE9
CURSL	0B1D	CURSR	0B02	CURSU	0AF9	D1M	0504	DATA	0991
DATA1	0A24	DCL	0CDE	DEL	0B5A	DEL0	0B64	DEL1	0B78
DEL10	0B7D	DEL1M	0517	DEL2	0B88	DEL50M	04AA	DEL6	0511
DELT	0500	DISPM	071E	DLY	055B	DLYR	0554	DPRNT	11F2
DSCL	0C6C	DSMAG	0C1B	DSP0	0844	DSP1	0852	DSP3	0866
DSP4	0869	DSPJR	085C	DSPNAM	05CF	DSPR	0C3E	DSPRED	0C3E
DSPW	0C50	DSPWRR	0C49	DSPXY	11D1	DTADR	1154	DUMP	0120
DUMP0	012A	DUMP1	012F	DWLDIR	0C5D	DWLDRN	0C66	DYSCSL	06D6
EDGE	0446	EDGE1	044E	EHL	0033	ESET	0583	EXADR	1156
EXIT	00B1	F00	0816	FOARE	0023	FARE	1200	FFWD	04E1
FKAE	11F0	FLASH	000D	FLASW	0829	FLPOS	0003	FLSDAT	000E
FNCOM	01C0	FOUMES	0241	FTAB	07C0	FUNC	0744	GAP	03C7
GAP1	03D7	GAP2	03DF	GAP3	03E5	GETKY	0832	GETL	06A4
GETLA	078A	GETLBR	05C5	GETLC	0799	GETLKN	0791	GETLR	0761
GLOP1	07AC	GLOP2	07B6	GOOUT	00AE	GRAPH	0A98	GRPHO	098B
GT2	0732	GT5	0751	GTBRK	0756	GTCR	0769	GTCR0	0779
HEX	05FD	HEX1	060F	HEXCR	0612	HL1	0621	HLHEX	0614
HOCT	0FE4	HOME	0B52	IBUFE	1140	INIC1	0016	INST	0BA1
INST0	0BB8	INST1	0BD6	INST2	0BCB	INSTRO	0BAE	IOTBL	0069
JST1	01BD	JUMP	024B	KADD	09CF	KANA	0A7A	KANA0	0982
KATM	09D0	KBSET	0571	KD1	09A9	KDATW	0026	KDATW1	0027
KDIN	099E	KEY	0926	KEYDIS	06ED	KEYFL	06DA	KEYKEY	090F
KEYW	06AE	KEYW1	06C4	KEYW2	06BD	KEYW3	06CE	KFINO	073B
KFINA	098E	KFINB	0A1E	KGRP	09D7	KIN	058C	KIN1	058F
KIN2	01A8	KINP	0598	KKANA	09D2	KMODE	11D0	KNUMB	06A3
KNUMBS	06A2	KSKA1	0A17	KSKA2	0A21	KSKANA	0A12	KSML	09ED
KSTD	11F4	KSWEP	0926	KTBL	0D1E	KTBLG	0DAE	KTBLK	0DCE
KTBLKS	0E36	KTBLR	0E06	KTBLS	0D76	KYBDA	11F3	KYFL1	06DE
KYFL2	06E6	LAMODE	0A6C	LETNL	0A2E	LMONLY	09CC	LOAMES	022D
LONG	0539	M#TBL	1002	MAGA	0C20	MANG	11D3	MCLECT	00F9
MELDY	0F3F	MENAME	0153	MLD1	0F47	MLD2	0F7A	MLD3	0F82
MLD4	0F86	MLOAD	01CB	MLOVE	01CF	MNAM1	0188	MONIT	0000
MOT1	046B	MOT2	0478	MOTOR	0457	MOTW	047D	MOTWG	0497
MPLAY	04D2	MR	00FF	MRUN	07DE	MSAVE	014E	MSG	0889
MSG1	088C	MSG2	0898	MSGX	087B	MSGX1	087E	MSTOP	04CE
MTBL	0FEA	MVERY	021A	MVRFY	0217	MWARK	0216	NAME	1141
NAMECK	01F6	NL	0A29	NLMSG	05B6	NLPHLS	05BC	NMSGST	0228
NOKD	0967	NOKD1	0967	NOKD2	096F	NOKKEY	0901	NOMLK	09C0
OCTV	11EF	OKMES	0248	ONP1	0F8D	ONP2	0F9A	ONP3	0FB2
ONPU	0F8A	ONTYO	0005	OPEN	048C	OPTBL	101A	OUTRT	0A71
PLAY	049B	PON1	0C37	PORET1	0F87	PRNT	08C6	PRNTS	08C4
PRNTT	086C	PRT2	08B0	PRT3	08B5	PRT4	08B8	PRTHL	05D8
PRTHX	05DD	RATIO	0036	RBY1	03A4	RBY2	03BC	RBY3	03C5
RBYTE	03A0	RD1	0297	RD2	02A6	RDDAT	02B2	RDEL	107F
RDINF	028E	REGIST	0D07	REPT	0719	REPT1	0709	REPT2	0714
REPTCT	0035	RETHB	02FF	RETHB1	03EE	RETN	0A75	ROT	0995
ROTE	099B	RTAP1	030F	RTAP2	0327	RTAP3	034A	RTAPE	030B
RVGRP	0A09	RYTHM	1033	RYTHM1	104B	RYTHM2	1067	RYTHM3	1058

```
SAME    063A  SAME1   063D  SAME2   0643  SAME3   0647  SAVEGO  01B5
SCRDSD  0C12  SCREND  000C  SCRN    D000  SCRO2   0AE0  SCROL   0AA7
SCROST  000B  SCRSET  08D3  SCRSIZ  001B  SCRST   0013  SCRSTD  0C15
SERSP   04B1  SETMES  0672  SHIFT   09F2  SHIFT1  0A01  SHL     002B
SHORT   051D  SIZE    1152  SMALL   0A89  SMALLO  0988  SOUT    0F22
SOUT1   0F24  SOUT2   0F35  SS      00A9  SSET    057A  SSP1    04BD
ST      00B1  STACK   002E  START   003B  START2  008F  STPRET  02AD
STRGF   002D  SUMDT   002B  SUMMES  0692  SWEP    093C  SWRK    0015
SYOKI   00EB  TAB     07F0  TAB1    07FF  TAB2    0806  TABDAT  11C0
TABLE1  1024  TAPER   034D  TDPCT   0A4E  TEMPW   001D  TIMRD   0EA9
TIMST   0E5E  TITMES  1093  TM1     03FA  TM2     03FB  TM3     040D
TM4     0421  TMARK   03F1  TMR1    0EFA  TMS1    0E77  TMUP    0F07
TMX     0ED2  TMX1    0F01  TMX2    0EE7  TOF     0007  TSPE    04F9
TVF1    035C  TVF2    036A  TVRFY   0358  VERFY   02BE  VERMES  0236
WBY1    0395  WBYTE   038F  WPRMES  0684  WRDAT   0282  WRI1    025A
WRI2    0271  WRIMES  067B  WRINF   0251  WTAP1   02DE  WTAP2   02E7
WTAP4   0302  WTAPE   02DA  XTEMP   0E50
```

A.5 BASICテープのコピー作成と改訂の方法

BASICテープのコピーを作成したり、BASICインタープリタMZ-1Z001あるいはMonitor MZ-1Z001Mの改訂を行う場合に必要な操作をここにまとめておきます。

MZ-2000に付属のカセットテープ「BASIC MZ-1Z001」は、テープの寿命を超えた使用あるいは不慮の事故による破損等によって使用できなくなることが考えられますので、ここで述べるコピー操作に従ってあらかじめ予備のBASICテープを作成しておくようお願い致します。また、万一、BASICインタープリタもしくは、Monitorプログラム内にバグ（プログラムの誤り：bug）が見つかった場合は、もよりのシャープサービス窓口（技術サービス部・サービスステーション・サービスブランチ）へお問い合わせください。訂正方法をお知らせいたします。

なお、主要MZマイコンショップへは、適時「MZソフトウェア・インフォメーション」を配布しておりますのでご利用ください。

■ BASICテープのコピーの作り方†)

(1) BASICインタープリタMZ-1Z001を起動します。

(2) MONコマンドを実行して、システムコントロールをMonitor MZ-1Z001Mへ移します。

(3) モニタコマンドM（Memory correction）によって、次のようにMonitorプログラム自身の一部分を変更します。

アドレス	変更前のデータ		変更後のデータ
0289	2A	→	21
028A	54	→	00
028B	11	→	80
02B9	2A	→	21
02BA	54	→	00
02BB	11	→	80

(4) 続いて、モニタコマンドL（Load）によって、BASICテープをロードします。この時、BASICインタープリタとMonitorプログラムは、(3)でMonitorプログラムの変更を行ったために、メモリエリア8000H番地以降にロードされます。
ローディングを終了したら、新しいカセットテープをカセットデッキにセットし巻き戻しを実行しておきます。

(5) 次に、モニタコマンドJ（Jump）によって、コントロールを、01B5H番地へ移します。この操作を行うと、自動的にカセットテープへの書き込み作業が開始され、BASIC（およびMonitorプログラム）のコピーテープが作成されます。

(6) 作成したコピーテープは、書き込み禁止用のツメを折ってライトプロテクトをかけておき大切に保管してください。

(7) Monitorプログラムあるいは BASICインタープリタでプログラミングを再開する際には、(3)で行ったのと逆の操作によって、変更したデータをもとのデータに戻さなければなりません。

†) モニタコマンドSによって、Monitorプログラム自身を含むエリアをカセットテープにセーブしようとすると、サムチェックでエラーが発生します。従って、BASICテープのコピーは、単純にモニタコマンドSで作成することができず、ここで示す操作が必要となるのです。

■ **BASICインタープリタMZ-1Z001またはMonitor MZ-1Z001Mの改訂方法**

BASICインタープリタMZ-1Z001または、Monitor MZ-1Z001Mの改訂を行うには、前記のコピー操作に於いて、(4)のローディング操作と(5)のセーブ操作の間に、データの変更（改正）操作を加えることになります。